Nondeterministic Modal Interfaces☆,☆☆

Ferenc Bujtor[a], Sascha Fendrich[b], Gerald Lüttgen[b], Walter Vogler[a]

[a] *Institut für Informatik, University of Augsburg, Germany*
[b] *Software Technologies Research Group, University of Bamberg, Germany*

Abstract

Interface theories are employed in the component-based design of concurrent systems. They often emerge as combinations of Interface Automata (IA) and Modal Transition Systems (MTS), e.g., Nyman et al.'s IOMTS, Bauer et al.'s MIO, Raclet et al.'s MI or our MIA. In this paper, we generalise MI to *nondeterministic* interfaces, for which we properly resolve the longstanding conflict between unspecified inputs being allowed in IA but forbidden in MTS. With this solution we achieve, in contrast to related work, an *associative* parallel composition, a *compositional* preorder, a conjunction on interfaces with *dissimilar alphabets* supporting perspective-based specifications, and a quotienting operator for decomposing *nondeterministic* specifications in a single theory. In addition, we define a hiding and a restriction operator, complement conjunction with a disjunction operator and illustrate our interface theory by means of a simple example.

Keywords: Interface Theories, Modal Interface Automata, Component Based Design, Modal Transition Systems, Disjunctive Must-Transitions

1. Introduction

Interface theories support the component-based design of concurrent systems and offer a semantic framework for, e.g., software contracts [2] and web services [3]. Several such theories are based on de Alfaro and Henzinger's *Interface Automata* (IA) [4], whose distinguishing feature is a parallel composition on labelled transition systems with inputs and outputs, where receiving an unexpected input is regarded as an error, i.e., a communication mismatch. In so-called *pessimistic* interface theories [5], a parallel composition of components is not defined, if such a mismatch occurs. In *optimistic* theories [6, 7, 8, 9, 10],

☆An extended abstract of this paper appeared in [1].

☆☆Research support was provided by the DFG (German Research Foundation) under grants LU 1748/3-1 and VO 615/12-1.

Email addresses: `ferenc.bujtor@informatik.uni-augsburg.de` (Ferenc Bujtor), `sascha.fendrich@swt-bamberg.de` (Sascha Fendrich), `gerald.luettgen@swt-bamberg.de` (Gerald Lüttgen), `walter.vogler@informatik.uni-augsburg.de` (Walter Vogler)

such as the ones we consider here, a communication mismatch is acceptable as long as the system environment prevents that it can be reached; technically, all those states of the parallel composition are pruned from which entering an error state cannot be prevented by any so-called *helpful* environment.

Various researchers have combined IA with Larsen's *Modal Transition Systems* (MTS) [11], featuring may- and must-transitions to express allowed and required behaviour, resp. In a refinement of an interface, all required behaviour must be preserved and no disallowed behaviour may be added. Whereas in IA outputs are optional, they may now be enforced in theories combining IA and MTS, such as Nyman et al.'s IOMTS [8], Bauer et al.'s MIO [5], Raclet et al.'s *Modal Interfaces* (MI) [10] and our *Modal Interface Automata* (MIA) [9, 12]. In this article we extend MI to nondeterministic systems, yielding the most general approach to date and permitting new applications, since nondeterminism arises, e.g., from races in networks. We build upon our prior work in [12], from which we adopt disjunctive must-transitions, which are needed for operationally defining conjunction on interfaces. Conjunction is a key operator in interface theories, supporting perspective-based specification and corresponding to the greatest lower bound wrt. refinement. We also consider the dual disjunction operator.

Combining IA and MTS is, however, problematic due to a conflict between unspecified inputs being forbidden in MTS but allowed in IA with arbitrary behaviour afterwards. In IOMTS [8], the MTS-view was adopted and, as a consequence, compositionality of refinement wrt. the parallel operator was lost. In [12] we followed the IA-view but found that reconciling the two views is essential for a more flexible conjunction. Flexibility is needed regarding the alphabets of the conjuncts that are to be composed; intuitively, each conjunct models a different perspective (i.e., a single system requirement) that only refers to the actions relevant to that perspective.

Here, we propose a middle way to reconcile the IA- and MTS-views by adding the option to treat an input i in a state p according to the IA-approach: If i should be allowed with subsequent arbitrary behaviour, we add an i-may-transition from p to a special *universal* state e that can be refined in any way. We need this option, in particular, when defining parallel composition. In contrast, if there is no i-transition originating in p, then i is forbidden in p according to the MTS-view. The idea behind e is similar to the one presented for MI in [10], where an ordinary state that has a may-loop for each action is added to a parallel composition. This way, however, associativity of parallel composition is lost. We avoid this problem since e is treated specially in our notion of refinement, which has far reaching consequences for many of the proofs; see Sec. 3.2 for a more detailed discussion of e. Now, with the universal state e and unlike the approach in [9, 12], our interface theory, which we continue to call MIA, allows for a proper distinction between may- and must-transitions for inputs. This enables us to define the desired, more flexible conjunction using a simple alphabet extension mechanism in the sense of [10].

Our proposed reconciliation results in an interface theory that generalises the fully deterministic MI, where also internal actions are forbidden, to *nondeterministic* interfaces. Unlike IA and our previous work [9, 12], we also do away with determinism on input-transitions. As in MI, our MIA theory is equipped with a multicast parallel composition, where one output can synchronise with several inputs. This is accompanied by hiding and restriction operators for scoping actions [13, 14]. Parallel composition and hiding together (cf. [15]) are more expressive than the binary parallel composition of IA used in [5, 8, 9, 12]. We also develop a quotienting operator $/\!\!/$ as a kind of inverse of parallel composition $\|$. For a specification P and a given component D, quotienting constructs the most general component Q such that $Q\|D$ refines P. Quotienting is a practical operator: it can be used for decomposing concurrent specifications stepwise, specifying contracts [16] and reusing components. In contrast to [10], our quotienting permits *nondeterministic* specifications and complements $\|$ rather than a simpler parallel product without pruning.

In summary, our new interface theory MIA generalises and improves upon existing theories combining IA and MTS: parallel composition is commutative and associative (cf. Sec. 3), quotienting also works for nondeterministic specifications (cf. Sec. 4), conjunction properly reflects perspective-based specification (cf. Secs. 5 and 6), and refinement (cf. Sec. 2) is compositional and permits alphabet extension (cf. Sec. 6). We illustrate the utility of MIA by means of a simple example (cf. Sec. 7).

2. Modal Interface Automata: The Setting

In this section we define *Modal Interface Automata* (MIA) and the supported operations. Essentially, MIAs are state machines with disjoint input and output alphabets, as in Interface Automata (IA) [4], and two transition relations, *may* and *must*, as in Modal Transition Systems (MTS) [11]. May-transitions describe permitted behaviour, while must-transitions describe required behaviour. Unlike previous versions of MIA [9, 12] and also unlike other similar theories, we introduce a special *universal state* e capturing arbitrary behaviour.

Definition 1 (Modal Interface Automata)**.** *A* Modal Interface Automaton *(MIA) is a tuple* $(P, I, O, \longrightarrow, \dashrightarrow, p_0, e)$*, where*

- *P is the set of states including the* initial state *p_0 and the* universal state *e,*
- *I and O are disjoint sets, the alphabets of* input *and* output actions*, not containing the special internal action τ, and $A =_{df} I \cup O$ is called the* alphabet*,*
- *$\longrightarrow \subseteq P \times (A \cup \{\tau\}) \times (\mathcal{P}(P) \setminus \emptyset)$ is the* disjunctive must-transition *relation, with $\mathcal{P}(P)$ being the powerset of P,*
- *$\dashrightarrow \subseteq P \times (A \cup \{\tau\}) \times P$ is the* may-transition *relation.*

We require the following conditions:

1. *For all $\alpha \in A \cup \{\tau\}$, $p \xrightarrow{\alpha} P'$ implies $\forall p' \in P'.\, p \overset{\alpha}{\dashrightarrow} p'$* (syntactic consistency)*,*

2. e appears in transitions only as the target state of input may-transitions (sink condition)*.*

A MIA P is called universal *if $P = (\{e\}, I, O, \emptyset, \emptyset, e, e)$ for alphabets I, O.*

Cond. 1 states that whatever is required should be allowed; this syntactic consistency is a natural and standard condition (see [11]). Regarding Cond. 2, recall that we use e to express that an input is optional in some state, with arbitrary behaviour afterwards. Note that there might very well be ordinary states without any outgoing transitions for some input i; in other words, a MIA does not have to be *input-enabled* like the IO-Automata in [15].

Observe that our disjunctive must-transitions have a single label, in contrast to Disjunctive MTS (DMTS) [17]. In the context of MTS, this is sufficient for intuitively and compactly representing (a) conjunction, as shown in [9], and (b) parallel composition, which would otherwise require an indirect definition via, e.g., Acceptance Automata [18, 19], as suggested in [20]. Our restriction to single labels does not seem to restrict the expressible sets of implementations, i.e., τ-free labelled transition systems (LTS), as studied by Fecher and Schmidt [21] and Beneš et al. [20], when – analogous to DMTS – allowing arbitrary sets of initial states in MIAs.

In the following we identify a MIA $(P, I, O, \longrightarrow, \dashrightarrow, p_0, e)$ with its state set P and, if needed, use index P when referring to one of its components, e.g., we write I_P for I. Similarly, we write, e.g., I_1 instead of I_{P_1} for MIA P_1. In addition, we let i, o, a, ω and α stand for representatives of the alphabets I, O, A, $O \cup \{\tau\}$ and $A \cup \{\tau\}$, resp.; we write $A = I/O$ when highlighting inputs I and outputs O in an alphabet A. In the context of weak transitions, we use the notation $\hat{\alpha}$, where $\hat{\alpha} =_{\mathrm{df}} a$ if $\alpha = a \neq \tau$ and $\hat{\alpha} =_{\mathrm{df}} \varepsilon$ if $\alpha = \tau$. Furthermore, outputs and internal actions are called *local* actions since they can be controlled locally by P. For notational convenience, we let $p \xrightarrow{a} p'$, $p \not\xrightarrow{a}$ and $p \overset{a}{\not\dashrightarrow}$ denote $p \xrightarrow{a} \{p'\}$, $\nexists P'.\ p \xrightarrow{a} P'$ and $\nexists p'.\ p \overset{a}{\dashrightarrow} p'$, resp. In figures, we often refer to an action a as $a?$ if $a \in I$, and $a!$ if $a \in O$. Must-transitions (may-transitions) are drawn using solid, possibly splitting arrows (dashed arrows); any depicted must-transition also implicitly represents the underlying may-transition(s) due to syntactic consistency.

We now define *weak* must- and may-transition relations that abstract from transitions labelled by τ, as is needed for MIA refinement. It is an alternative, more general definition than the one presented in [12]. In [12] and [1], we have failed to notice that our conjunction operator applied to infinite MIAs can result in infinite target sets of disjunctive must-transitions (Rules (OMust), (IMust) in Def. 32; see p. 34 for an example of this). Consequently, we now allow such target sets in Def. 1 above. As a consequence, we modify also the definition of weak transitions; in order to derive adequate weak must-transitions, they are built up back-to-front.

Definition 2 (Weak Transition Relations)**.** *For an arbitrary MIA P, we define* weak *must- and may-transition relations, $\Longrightarrow$ and $=\!=\!\Rightarrow$ resp., as the smallest*

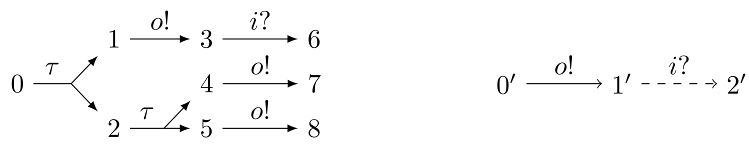

Figure 1: Examples of weak transitions and refinement.

relations satisfying the following conditions, where we write $P' \overset{\hat{\alpha}}{\Longrightarrow} P''$ *as a shorthand for* $\forall p \in P' \, \exists P_p.\, p \overset{\hat{\alpha}}{\Longrightarrow} P_p$ *and* $P'' = \bigcup_{p \in P'} P_p$*:*

1. $p \overset{\varepsilon}{\Longrightarrow} \{p\}$ *for all* $p \in P$,
2. $p \overset{\tau}{\longrightarrow} P'$ *and* $P' \overset{\hat{\alpha}}{\Longrightarrow} P''$ *implies* $p \overset{\hat{\alpha}}{\Longrightarrow} P''$,
3. $p \overset{a}{\longrightarrow} P'$ *and* $P' \overset{\varepsilon}{\Longrightarrow} P''$ *implies* $p \overset{a}{\Longrightarrow} P''$,
4. $p \overset{\varepsilon}{=\!\Rightarrow} p$,
5. $p \overset{\varepsilon}{=\!\Rightarrow} p'' \overset{\tau}{\dashrightarrow} p'$ *implies* $p \overset{\varepsilon}{=\!\Rightarrow} p'$,
6. $p \overset{\varepsilon}{=\!\Rightarrow} p'' \overset{\alpha}{\dashrightarrow} p''' \overset{\varepsilon}{=\!\Rightarrow} p'$ *implies* $p \overset{\alpha}{=\!\Rightarrow} p'$.

We write $\overset{a}{\longrightarrow}\overset{\varepsilon}{\Longrightarrow}$ *for transitions that are built up according to Case 3 and call them* trailing-weak *must-transitions. Similarly,* $\overset{a}{\dashrightarrow}\overset{\varepsilon}{=\!\Rightarrow}$ *stands for trailing-weak may-transitions.*

For examples of weak transitions, consider the MIA on the left-hand side of Fig. 1. By applying Def. 2.1 and 2.2, any τ-transition is also a weak ε-transition. Similarly, every a-transition is also a weak a-transition by Def. 2.1 and 2.3. Transition $2 \overset{\tau}{\longrightarrow} \{4, 5\}$ can be extended to $2 \overset{o}{\Longrightarrow} \{7, 8\}$ by applying Def. 2.2. Hence, $0 \overset{\tau}{\longrightarrow} \{1, 2\}$ extends to $0 \overset{o}{\Longrightarrow} \{3, 7, 8\}$. Observe that our weak must-transitions correspond to standard weak transitions of LTS in the case that only must-transitions with a single target state are used.

When reasoning about weak must-transitions, e.g., in Lems. 3 and 21 below, we consider a derivation of a weak must-transition according to Def. 2 as a tree and each node as being larger than the nodes from which it is derived. Although the tree might be infinitely branching, larger-than is a Noetherian partial order. Hence, one can apply (Noetherian, transfinite) induction on the derivation of a weak must-transition.

Lemma 3. *Consider an arbitrary MIA* P*.*

(a) $p \overset{\varepsilon}{\Longrightarrow} \bar{P}$ *and* $\bar{P} \overset{\hat{\alpha}}{\Longrightarrow} P'$ *implies* $p \overset{\hat{\alpha}}{\Longrightarrow} P'$,

(b) $p \overset{a}{\Longrightarrow} \bar{P}$ *and* $\bar{P} \overset{\varepsilon}{\Longrightarrow} P'$ *implies* $p \overset{a}{\Longrightarrow} P'$.

Proof. (a) We proceed by induction on the definition of $p \overset{\varepsilon}{\Longrightarrow} \bar{P}$. Regarding Def. 2.1, the claim is trivial. Now assume that $p \overset{\varepsilon}{\Longrightarrow} \bar{P}$ is due to Def. 2.2, i.e., we have $p \overset{\tau}{\longrightarrow} P''$ and, for each $p'' \in P''$, there is some $\bar{P}_{p''}$ with $p'' \overset{\varepsilon}{\Longrightarrow} \bar{P}_{p''}$ and $\bar{P} = \bigcup_{p'' \in P''} \bar{P}_{p''}$. By premise $\bar{P} \overset{\hat{\alpha}}{\Longrightarrow} P'$, some $P_{\bar{p}}$ exists for each $\bar{p} \in \bar{P}$ such that $\bar{p} \overset{\hat{\alpha}}{\Longrightarrow} P_{\bar{p}}$ and $P' = \bigcup_{\bar{p} \in \bar{P}} P_{\bar{p}}$. For each $p'' \in P''$, $P'_{p''} =_{\mathrm{df}} \bigcup_{\bar{p} \in \bar{P}_{p''}} P_{\bar{p}}$ satisfies $\bar{P}_{p''} \overset{\hat{\alpha}}{\Longrightarrow} P'_{p''}$ and, by induction hypothesis, $p'' \overset{\hat{\alpha}}{\Longrightarrow} P'_{p''}$. By Def. 2.2, this implies $p \overset{\hat{\alpha}}{\Longrightarrow} \bigcup_{p'' \in P''} P'_{p''}$. This target set is clearly the union of some $P_{\bar{p}}$ with $\bar{p} \in \bar{P}$; moreover, each $\bar{p} \in \bar{P}$ is in some $\bar{P}_{p''}$, and the target set covers $P'_{p''} \supseteq P_{\bar{p}}$. Hence, the target set is P' and we are done. The case of Def. 2.3 does not apply.

(b) Similarly to (a), we apply induction on the derivation of $p \overset{a}{\Longrightarrow} \bar{P}$. Case 1 of Def. 2 does not apply. Case 2 is shown as above, observing that we need $p'' \overset{a}{\Longrightarrow}$ twice, $\bar{p} \overset{\varepsilon}{\Longrightarrow}$, $\bar{P}_{p''} \overset{\varepsilon}{\Longrightarrow}$ and $p \overset{a}{\Longrightarrow}$. Case 3 is also similar to Case 2 in (a), except that all weak transitions not originating in p are labelled ε, and we use (a) instead of the induction hypothesis. $\square$

Now we define our simulation-based refinement relation. It is a weak alternating simulation that is conceptually similar to the observational modal refinement found, e.g., in [22]. A notable aspect, originating from IA [4], is that inputs must be matched immediately, i.e., only trailing τs are allowed. Intuitively, this is because of the requirement that a signal sent from one system must be received immediately; otherwise, it is considered an error (communication mismatch). Since one wishes not to introduce new errors during refinement, a refined system must immediately provide all specified inputs. This is discussed further in Remark 9.

We treat the universal state e as completely underspecified, i.e., any state refines it; this is only possible since e is not an ordinary state. Recall that we have an i-may-transition from some state p to e to express that, like in the IA-approach, p can be refined by a state with an i-transition followed by arbitrary behaviour. We define our refinement preorder for MIAs with common input and output alphabets here and relax this restriction in Sec. 6.

Definition 4 (MIA Refinement)**.** *Let P, Q be MIAs with common input and output alphabets. A relation $\mathcal{R} \subseteq P \times Q$ is a* MIA-refinement relation *if, for all $(p, q) \in \mathcal{R}$ with $q \neq e_Q$, the following conditions hold:*

(i) $p \neq e_P$,

(ii) $q \overset{i}{\longrightarrow} Q'$ *implies* $\exists P'.\, p \overset{i}{\longrightarrow}\overset{\varepsilon}{\Longrightarrow} P'$ *and* $\forall p' \in P'\, \exists q' \in Q'.\, (p', q') \in \mathcal{R}$,

(iii) $q \overset{\omega}{\longrightarrow} Q'$ *implies* $\exists P'.\, p \overset{\hat{\omega}}{\Longrightarrow} P'$ *and* $\forall p' \in P'\, \exists q' \in Q'.\, (p', q') \in \mathcal{R}$,

(iv) $p \overset{i}{\dashrightarrow} p'$ *implies* $\exists q'.\, q \overset{i}{\dashrightarrow}\overset{\varepsilon}{\Rightarrow} q'$ *and* $(p', q') \in \mathcal{R}$,

(v) $p \overset{\omega}{\dashrightarrow} p'$ *implies* $\exists q'.\, q \overset{\hat{\omega}}{\Rightarrow} q'$ *and* $(p', q') \in \mathcal{R}$.

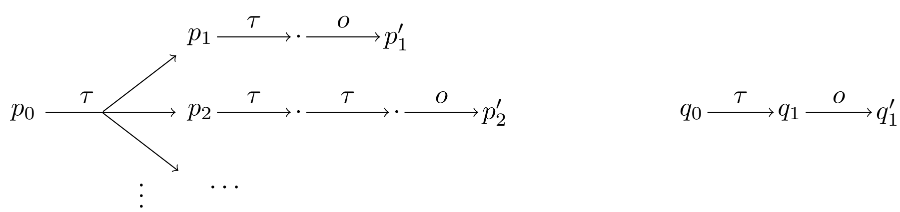

Figure 2: Example of refining a weak transition.

We write $p \sqsubseteq q$ and say that p MIA-refines *q, if there exists a MIA-refinement relation $\mathcal{R}$ such that $(p,q) \in \mathcal{R}$, and we let $p \sqsupseteq\sqsubseteq q$ stand for $p \sqsubseteq q$ and $q \sqsubseteq p$. Furthermore, we extend these notations to MIAs and write $P \sqsubseteq Q$ if $p_0 \sqsubseteq q_0$ and use $\sqsupseteq\sqsubseteq$ analogously.*

An example of a refinement can be found in Fig. 1, where the left MIA refines the right one due to the refinement relation $\{(0,0'),(1,0'),(2,0'),(4,0'),(5,0'),(3,1'),(7,1'),(8,1'),(6,2')\}$. Observe how the refined states 3 and 7 (and 8) of state $1'$ implement the outgoing $i?$-may-transition differently.

For another example, consider the weak transition $p_0 \stackrel{o}{\Longrightarrow} P' =_{\mathrm{df}} \{p'_1, p'_2, \ldots\}$ of 'finite but unbounded depth' in Fig. 2, which arises from our back-to-front definition (cf. Def. 2). This weak transition is intuitively justified, since each target of the disjunctive τ-must-transition guarantees o. Technically, each p_i refines q_1 and, hence, p_0 refines q_0 according to Def. 4. Therefore, $p_0 \stackrel{o}{\Longrightarrow} P'$ is needed to make Prop. 5 (iii) below true for $q_0 \stackrel{o}{\Longrightarrow} q'_1$.

As we show next, Lem. 3 allows us to replace the transition in the premises of (ii) and (iii) above by a trailing weak and a weak one, resp.; the analogous replacement in (iv) and (v) is standard. This result is needed for proving that $\sqsubseteq$ is a preorder.

Proposition 5. *Let $\mathcal{R} \subseteq P \times Q$ be a MIA-refinement relation for MIAs P and Q, and let $(p,q) \in \mathcal{R}$ with $q \neq e_Q$. Then, the following generalisations of Def. 4(ii)–(v) hold:*

(ii) $q \stackrel{i}{\longrightarrow}\stackrel{\varepsilon}{\Longrightarrow} Q'$ implies $\exists P'.\, p \stackrel{i}{\longrightarrow}\stackrel{\varepsilon}{\Longrightarrow} P'$ and $\forall p' \in P'\, \exists q' \in Q'.\, (p',q') \in \mathcal{R}$,

(iii) $q \stackrel{\hat{\omega}}{\Longrightarrow} Q'$ implies $\exists P'.\, p \stackrel{\hat{\omega}}{\Longrightarrow} P'$ and $\forall p' \in P'\, \exists q' \in Q'.\, (p',q') \in \mathcal{R}$,

(iv) $p \stackrel{i}{\dashrightarrow}\stackrel{\varepsilon}{=\Rightarrow} p'$ implies $\exists q'.\, q \stackrel{i}{\dashrightarrow}\stackrel{\varepsilon}{=\Rightarrow} q'$ and $(p',q') \in \mathcal{R}$,

(v) $p \stackrel{\hat{\omega}}{=\Rightarrow} p'$ implies $\exists q'.\, q \stackrel{\hat{\omega}}{=\Rightarrow} q'$ and $(p',q') \in \mathcal{R}$.

Proof. The proofs of Parts (iv) and (v) are standard; the proof of Part (ii) is very similar to that of Part (iii), although the third case below is not relevant for Part (ii); thus, we focus on proving Part (iii) concerning weak disjunctive transitions. We proceed by induction on the definition of $q \stackrel{\hat{\omega}}{\Longrightarrow} Q'$:

- Let $\omega = \tau$ and $Q' = \{q\}$. Then, we choose $P' =_{\mathrm{df}} \{p\}$.

- Let $q \xrightarrow{\tau} \bar{Q}$ and $\forall \bar{q} \in \bar{Q}\, \exists Q_{\bar{q}}.\, \bar{q} \stackrel{\hat{\omega}}{\Longrightarrow} Q_{\bar{q}}$ with $Q' = \bigcup_{\bar{q} \in \bar{Q}} Q_{\bar{q}}$ according to Def. 2.2. By assumption, a weak transition $p \stackrel{\varepsilon}{\Longrightarrow} \bar{P}$ with $\forall \bar{p} \in \bar{P}$ $\exists \bar{q} \in \bar{Q}.\, (\bar{p}, \bar{q}) \in \mathcal{R}$ exists. Choosing for each $\bar{p} \in \bar{P}$ a suitable $\bar{q}$, we get some $P_{\bar{p}}$ such that $\bar{p} \stackrel{\hat{\omega}}{\Longrightarrow} P_{\bar{P}}$ and $\forall p' \in P_{\bar{p}}\, \exists q' \in Q_{\bar{q}}.\, (p', q') \in \mathcal{R}$ by induction hypothesis. By Lem. 3(a), we obtain $p \stackrel{\hat{\omega}}{\Longrightarrow} P' =_{\mathrm{df}} \bigcup_{\bar{p} \in \bar{P}} P_{\bar{p}}$.

- Let $q \stackrel{\hat{\omega}}{\Longrightarrow} Q'$ due to Def. 2.3, i.e., $\hat{\omega} = o$, $q \xrightarrow{o} \bar{Q}$, $\forall \bar{q} \in \bar{Q}.\, \bar{q} \stackrel{\varepsilon}{\Longrightarrow} Q_{\bar{q}}$ and $Q' = \bigcup_{\bar{q} \in \bar{Q}} Q_{\bar{q}}$. The proof then proceeds as in the previous case, using Lem. 3(b). □

Corollary 6. *MIA refinement $\sqsubseteq$ is a preorder and the largest MIA-refinement relation.*

Proof. Reflexivity immediately follows from the fact that the identity relation on states is a MIA-refinement relation. For transitivity one shows that the composition of two MIA-refinement relations is again a MIA-refinement relation, using Prop. 5 and following the lines of [14]. The second claim follows since MIA-refinement relations are easily seen to be closed under union. □

3. Parallel Composition and Hiding

Interface Automata (IA) [23, 4] are equipped with an interleaving parallel operator, where an action occurring as an input in one interface is synchronised with the same action occurring as an output in some other interface; the synchronised action is hidden, i.e., labelled by τ. Since our work builds upon Modal Interfaces (MI) [10] we instead consider here a parallel composition, where the synchronisation of an interface's output action involves all concurrently running interfaces that have the action as input. Moreover, we include a separate operator for hiding outputs (cf. [15]). This properly generalises the binary communication of IA to multicast in MIA.

3.1. Parallel Composition

We present a parallel operator $\|$ on MIA in the same way as we did in [9, 12], except that common actions are not hidden immediately. Parallel composition is defined in two stages, similarly as in IA. First, a standard product $\otimes$ between two MIAs is introduced. Then, errors are identified, i.e., states where an output is not matched by an appropriate input, and all states from which reaching an error cannot be prevented are *pruned*, i.e., removed.

Definition 7 (Parallel Product)**.** *MIAs P_1, P_2 are* composable *if $O_1 \cap O_2 = \emptyset$. For such MIAs we define the* product *$P_1 \otimes P_2 = ((P_1 \times P_2) \cup \{e_{12}\}, I, O, \longrightarrow,$ $\dashrightarrow, (p_{01}, p_{02}), e_{12})$, where e_{12} is a fresh state, $I =_{df} (I_1 \cup I_2) \setminus (O_1 \cup O_2)$ and $O =_{df} O_1 \cup O_2$, and where $\longrightarrow$ and $\dashrightarrow$ are the least relations satisfying the following rules:*

$$
\begin{array}{llll}
\text{(PMust1)} & (p_1,p_2) \xrightarrow{\alpha} P_1' \times \{p_2\} & \textit{if} & p_1 \xrightarrow{\alpha} P_1' \textit{ and } \alpha \notin A_2 \\
\text{(PMust2)} & (p_1,p_2) \xrightarrow{\alpha} \{p_1\} \times P_2' & \textit{if} & p_2 \xrightarrow{\alpha} P_2' \textit{ and } \alpha \notin A_1 \\
\text{(PMust3)} & (p_1,p_2) \xrightarrow{a} P_1' \times P_2' & \textit{if} & p_1 \xrightarrow{a} P_1' \textit{ and } p_2 \xrightarrow{a} P_2' \textit{ for some } a \\
\text{(PMay1)} & (p_1,p_2) \overset{\alpha}{\dashrightarrow} (p_1',p_2) & \textit{if} & p_1 \overset{\alpha}{\dashrightarrow} p_1' \textit{ and } \alpha \notin A_2 \\
\text{(PMay2)} & (p_1,p_2) \overset{\alpha}{\dashrightarrow} (p_1,p_2') & \textit{if} & p_2 \overset{\alpha}{\dashrightarrow} p_2' \textit{ and } \alpha \notin A_1 \\
\text{(PMay3)} & (p_1,p_2) \overset{a}{\dashrightarrow} (p_1',p_2') & \textit{if} & p_1 \overset{a}{\dashrightarrow} p_1' \textit{ and } p_2 \overset{a}{\dashrightarrow} p_2' \textit{ for some } a.
\end{array}
$$

From the parallel product, parallel composition is obtained by pruning, i.e., one removes errors and states leading up to them via local actions, so called *illegal* states. This also cuts all input transitions leading to an illegal state.

In [24] we showed that de Alfaro and Henzinger have defined pruning in an inappropriate way in [23], such that associativity is violated. We remedied this by cutting not only an i-transition from some state p to an illegal state, but also all other i-transitions from p. Not only did we prove that this is correct, the solution is also intuitive since, this way, p describes the requirement that a helpful environment must not produce input i. This requirement is described in input-deterministic settings like [4] without any remedy.

Now, in [23, 24], p can be refined by a state with an i-transition and arbitrary behaviour afterwards. As explained above, we express this by introducing an i-may-transition to the universal state. This construction is necessary to achieve compositionality and associativity for parallel composition; see Fig. 10 in [9] for the compositionality flaw in IOMTS [8] and Fig. 4 for the associativity problem in MI [10], resp.

Definition 8 (Parallel Composition)**.** *Given a parallel product $P_1 \otimes P_2$, a state (p_1,p_2) is a* new error *if there is some $a \in A_1 \cap A_2$ such that (a) $a \in O_1$, $p_1 \overset{a}{\dashrightarrow}$ and $p_2 \overset{a}{\not\longrightarrow}$, or (b) $a \in O_2$, $p_2 \overset{a}{\dashrightarrow}$ and $p_1 \overset{a}{\not\longrightarrow}$. It is an* inherited error *if one of its components is a universal state, i.e., if it is of the form (e_1,p_2) or (p_1,e_2).*

We define the set $E \subseteq P_1 \times P_2$ of illegal *states as the least set such that $(p_1,p_2) \in E$ if (i) (p_1,p_2) is a new or inherited error or (ii) $(p_1,p_2) \overset{\omega}{\dashrightarrow} (p_1',p_2')$ and $(p_1',p_2') \in E$.*

Should the initial state be an illegal state, i.e., $(p_{01},p_{02}) \in E$, then e_{12} becomes the initial – and thus the only reachable – state of the parallel composition *$P_1 \parallel P_2$. In this case, P_1 and P_2 are called* incompatible.

Otherwise, $P_1 \parallel P_2$ is obtained from $P_1 \otimes P_2$ by pruning *illegal states as follows. If there is a state $(p_1,p_2) \notin E$ with $(p_1,p_2) \overset{i}{\dashrightarrow} (p_1',p_2') \in E$ for some $i \in I$, then all must- and may-transitions labelled i and starting at (p_1,p_2) are removed, and a single transition $(p_1,p_2) \overset{i}{\dashrightarrow} e_{12}$ is added. Furthermore, all states in E, all unreachable states (except for e_{12}) and all their incoming and outgoing transitions are removed. If $(p_1,p_2) \in P_1 \parallel P_2$, we write $p_1 \parallel p_2$ and call p_1 and p_2* compatible.

Observe that the parallel composition of MIAs results in a well-defined MIA. Firstly, this is true for the parallel product; in particular, e_{12} does not have any transitions at all. Secondly, pruning guarantees that all target sets of must-transitions are non-empty, and it preserves syntactic consistency and the sink

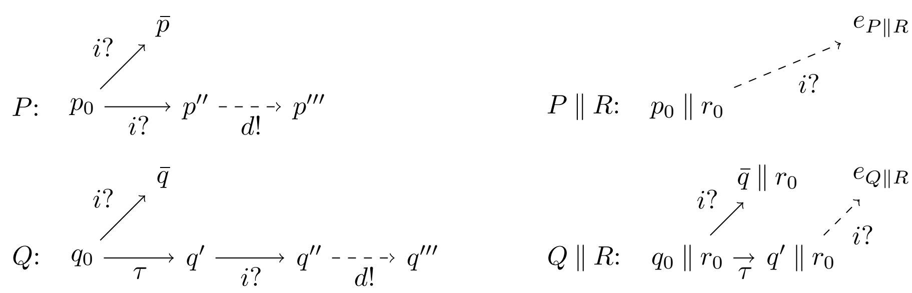

Figure 3: Illustration of the complications of pruning, where $A_P = A_Q = \{i\}/\{d\}$ and $A_R = \{d\}/\emptyset$.

condition. As an aside, even if we would not have required the sink condition in Def. 1, it would be enforced when applying parallel composition. Due to the universality of e, $P_1 \parallel P_2$ is universally refineable if P_1 and P_2 are incompatible.

Remark 9. *Recall that, in Def. 4, only trailing* τ*s are permitted when matching inputs. This is necessary for input must-transitions in order to avoid additional errors when refining a component; otherwise,* $\sqsubseteq$ *would not be a precongruence for parallel composition (cf. [4]). We now show that allowing leading* τ*s when matching input may-transitions would render our pruning insufficient. When generalising Def. 4(iv) this way, we would have* $P \sqsubseteq Q$ *in Fig. 3 due to* $\{(p_0, q_0), (\bar{p}, \bar{q}), (p_0, q'), (p'', q''), (p''', q''')\}$*. Their parallel compositions with* $R =_{df} (\{r_0, e_R\}, \{d\}, \emptyset, \emptyset, \emptyset, r_0, e_R)$ *would, with our current pruning, no longer be in the refinement relation:* $q_0 \parallel r_0$ *would still have an* i*-must-transition, while* $p_0 \parallel r_0$ *would have lost both* i*-must-transitions during pruning. Thus,* $\sqsubseteq$ *would not be a precongruence wrt. parallel composition.*

It is possible to repair this by a different pruning construction. For example, when cutting i*-transitions at some state* s*, one can go backward from* s *along* τ*-transitions and cut all outgoing* i*-transitions; in the example,* $q' \parallel r_0$ *has an* i*-transition that is cut and, consequently, we would also remove every* i*-transition originating from* $q_0 \parallel r_0$ *since* $q_0 \parallel r_0 \stackrel{\varepsilon}{\Longrightarrow} q' \parallel r_0$*. This different parallel composition fixes the current counterexample as it removes* $q_0 \parallel r_0 \xrightarrow{i} \bar{q} \parallel r_0$ *and its underlying may-transition, replacing them with an* i*-may-transition to the universal state. However, defining a general fix is much more involved since backward* and *forward propagation along* τ*s is necessary. This can be seen with a simple modification of the above example; just move the* d*-transition from state* p'' *to* $\bar{p}$ *and from* q'' *to* $\bar{q}$*.* □

In [10], Raclet et al. use a similar approach to pruning, but without an explicit universal state. Instead, when pruning illegal states, they introduce a state we denote as $\mathtt{tt}$, which almost behaves like our universal state. By construction, this state has only input may-transitions as incoming transitions. Furthermore, it has a may-loop for every action of the parallel composition so that it can be refined by any state, much like our universal state (see Def. 4(i)). But $\mathtt{tt}$

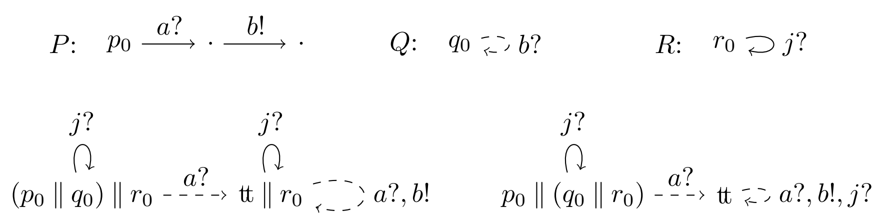

Figure 4: Differences between our state e and $\mathtt{tt}$ in [10], where $A_P = \{a\}/\{b\}$, $A_Q = \{b\}/\emptyset$ and $A_R = \{j\}/\emptyset$.

behaves differently in a parallel composition. To see this, consider the MIAs P, Q, R in Fig. 4, where we construct $(P \parallel Q) \parallel R$ according to [10]. Since $\mathtt{tt}$ is an ordinary state, it is combined with r_0 inheriting the j-must-loop. In our approach, the combination with r_0 is an inherited error, and the target state just has a j-may-loop.

More importantly, there is the severe problem that parallel composition in [10] is not associative. Consider $P \parallel (Q \parallel R)$, also shown in Fig. 4, which is not equivalent according to $\sqsupseteq\sqsubseteq$ (and the equivalence in [10]) to $(P \parallel Q) \parallel R$, due to the j-must-loop at $\mathtt{tt} \parallel r_0$. Note that our example does not rely on the multicast aspect of our parallel composition; it works just as well for the classic IA parallel composition.

We now prove that our parallel composition is indeed associative, starting with two lemmas.

Lemma 10. *If P, Q are composable MIAs, $p \parallel q \in P \parallel Q$, $o \in O_{P\parallel Q}$ and $i \in I_{P\parallel Q}$, then:*

1. *$p \parallel q \overset{o}{\dashrightarrow}$ iff $p \overset{o}{\dashrightarrow}$ and $o \in O_P$, or $q \overset{o}{\dashrightarrow}$ and $o \in O_Q$.*
2. *If $p \overset{i}{\not\longrightarrow}$ and $i \in I_P$ or if $q \overset{i}{\not\longrightarrow}$ and $i \in I_Q$, then $p \parallel q \overset{i}{\not\longrightarrow}$. The reverse implication does not hold in general.*

Proof. 1. Implication "$\Rightarrow$" is obvious. If implication "$\Leftarrow$" were false, then (p, q) would be a new error or $(p, q) \overset{o}{\dashrightarrow} (p', q')$ in $P \otimes Q$ with $p' \parallel q'$ undefined. Both would render (p, q) illegal and $p \parallel q$ undefined, leading to a contradiction.

2. This implication is also obvious, but the reverse implication does not hold since the must-transition of $p \parallel q$ might have been cut during pruning. $\square$

Lemma 11. *Given three MIAs P_1, P_2 and P_3, we have:*

1. *$(P_1 \parallel P_2) \parallel P_3$ is defined iff P_1, P_2 and P_3 are pairwise composable iff $P_1 \parallel (P_2 \parallel P_3)$ is defined as well.*
2. *$(P_1 \parallel P_2) \parallel P_3$ is equal to S obtained from applying pruning in one step to $(P_1 \otimes P_2) \otimes P_3$ (up to the name of the respective universal state). For this purpose, a state $((p_1, p_2), p_3)$ is a new error if, for some $i \neq j$ with*

$i, j \in \{1,2,3\}$, there is some $a \in A_i \cap A_j$ such that $a \in O_i$, $p_i \overset{a}{\dashrightarrow}$ and $p_j \overset{a}{\not\rightarrow}$; it is an inherited one, if $p_i = e_i$ for some $i \in \{1,2,3\}$.

Proof. 1. is easy. 2. For reasons of readability we use P, Q, R instead of P_1, P_2, P_3 and write (p,q,r) for $((p,q),r)$. Let E_{PQR} denote the illegal states of $(P \otimes Q) \otimes R$ as defined above when constructing S. We denote the illegal states of $P \otimes Q$ and $(P \| Q) \otimes R$ by E_{PQ} and $E_{(P\|Q)\otimes R}$ resp. Furthermore, let Err_{PQR}, Err_{PQ} and $Err_{(P\|Q)\otimes R}$ be the errors of the respective systems. We also say that two states p and q *produce an error*, if (p,q) is an error due to $p \overset{a}{\dashrightarrow}$ and $q \overset{a}{\not\rightarrow}$ while $a \in O_P \cap I_Q$ or vice versa.

Our first aim is to show that $E_{PQR} = (E_{PQ} \times R) \cup (E_{(P\|Q)\otimes R} \setminus (\{e_{P\|Q}\} \times R))$.

Part "$\subseteq$". We prove that $(p,q,r) \in E_{PQR}$ is contained in the r.h.s. by induction on the length of a local transition sequence from (p,q,r) to an error in Err_{PQR}. For the base case, we show $Err_{PQR} \subseteq (E_{PQ} \times R) \cup (E_{(P\|Q)\otimes R} \setminus (\{e_{P\|Q}\} \times R))$.

Consider $(p,q,r) \in Err_{PQR}$. If (p,q) is illegal in $P \otimes Q$ (this covers the cases that p or q is universal or that p and q produce an error), then $(p,q,r) \in E_{PQ} \times R$. Otherwise, $r = e_R$ and $(p,q,r) \in Err_{(P\|Q)\otimes R} \setminus (\{e_{P\|Q}\} \times R) \subseteq E_{(P\|Q)\otimes R} \setminus (\{e_{P\|Q}\} \times R)$, or r produces the error with p or q (or possibly both). W.l.o.g. let p and r produce the error because $p \overset{a}{\dashrightarrow}$ and $r \overset{a}{\not\rightarrow}$ for some $a \in O_P \cap I_R$ or because $p \overset{a}{\not\rightarrow}$ and $r \overset{a}{\dashrightarrow}$ for some $a \in I_P \cap O_R$. By Lem. 10.1, this leads to $p \| q \overset{a}{\dashrightarrow}$ and $r \overset{a}{\not\rightarrow}$ or, by Lem. 10.2, to $p \| q \overset{a}{\not\rightarrow}$ and $r \overset{a}{\dashrightarrow}$. Again, $(p,q,r) \in Err_{(P\|Q)\otimes R} \setminus (\{e_{P\|Q}\} \times R)$.

For the induction step, consider $(p,q,r) \in E_{PQR}$ such that $(p,q,r) \overset{\omega}{\dashrightarrow} (p',q',r') \in E_{PQR}$ and $(p',q',r') \in (E_{PQ} \times R) \cup (E_{(P\|Q)\otimes R} \setminus (\{e_{P\|Q}\} \times R))$ by induction hypothesis. By the argument at the beginning of the base case, we can assume that $p \| q$ is defined and, thus, $(p \| q, r)$ exists in $(P \| Q) \otimes R$. Thus, if $(p',q',r') \in E_{(P\|Q)\otimes R} \setminus (\{e_{P\|Q}\} \times R))$, then $(p,q,r) \in E_{(P\|Q)\otimes R} \setminus (\{e_{P\|Q}\} \times R))$ by the definition of E.

Finally, consider $(p',q',r') \in E_{PQ} \times R$. If the ω-transition is only performed by r, then $(p',q',r') = (p,q,r')$ and, thus, $(p,q) \in E_{PQ}$, contradicting that (p,q) is not illegal. Otherwise, if $\omega \in O_{P\otimes Q} \cup \{\tau\}$, then $(p,q) \overset{\omega}{\dashrightarrow} (p',q') \in E_{PQ}$ and $(p,q) \in E_{PQ}$, a contradiction. Thus, $\omega \in I_{P\otimes Q}$ and r performs ω as an output since, overall, it is an output. As $(p,q) \overset{\omega}{\dashrightarrow} (p',q') \in E_{PQ}$, this input transition is cut when pruning $P \otimes Q$, implying $p \| q \overset{\omega}{\not\rightarrow}$. This shows again that $(p,q,r) \in Err_{(P\|Q)\otimes R} \setminus (\{e_{P\|Q}\} \times R)$.

Part "$\supseteq$". We show that $(E_{PQ} \times R) \cup (E_{(P\|Q)\otimes R} \setminus (\{e_{P\|Q}\} \times R)) \subseteq E_{PQR}$.

First, we establish $E_{PQ} \times R \subseteq E_{PQR}$: We prove that $(p,q,r) \in E_{PQ} \times R$ is contained in E_{PQR} by induction on the length of a local transition sequence from (p,q) to an error in Err_{PQ}. In the base case $(p,q) \in Err_{PQ}$, we have that p and q produce an error or one of them is an error state. In either case $(p,q,r) \in Err_{PQR} \subseteq E_{PQR}$. For the induction step, consider some $(p,q) \overset{\omega}{\dashrightarrow} (p',q') \in E_{PQ}$ where, by induction hypothesis, $\{(p',q')\} \times R \subseteq E_{PQR}$. If $\omega \notin A_R$, then $(p,q,r) \stackrel{\omega}{\dashrightarrow} (p',q',r) \in E_{PQR}$, and we are done. If $\omega \in A_R$, then we must have $\omega \in I_R$. Now, either $(p,q,r) \in Err_{PQR}$ or $(p,q,r) \stackrel{\omega}{\dashrightarrow} (p',q',r') \in E_{PQR}$ for some r', and in either case we are done.

Second, we establish $E_{(P\|Q)\otimes R} \setminus (\{e_{P\|Q}\} \times R) \subseteq E_{PQR}$. We prove that $(p,q,r) \in E_{(P\|Q)\otimes R} \setminus (\{e_{P\|Q}\} \times R)$ is contained in E_{PQR} by induction on the length of a local transition sequence from $(p \parallel q, r)$ to an error in $Err_{(P\|Q)\otimes R}$. In the base case $(p \parallel q, r) \in Err_{(P\|Q)\otimes R} \setminus (\{e_{P\|Q}\} \times R)$, we have that $r = e_R$ and, thus, $(p,q,r) \in Err_{PQR} \subseteq E_{PQR}$, or that $p \parallel q$ and r produce an error. The latter means either $p \parallel q \stackrel{a}{\dashrightarrow}$ and $r \stackrel{a}{\not\longrightarrow}$ for some $a \in (O_P \cup O_Q) \cap I_R$, implying $p \stackrel{a}{\dashrightarrow}$ and $a \in O_P$ or $q \stackrel{a}{\dashrightarrow}$ and $a \in O_Q$ by Lem. 10.1, and hence $(p,q,r) \in Err_{PQR} \subseteq E_{PQR}$; or $p \parallel q \stackrel{a}{\not\longrightarrow}$ and $r \stackrel{a}{\dashrightarrow}$ for some $a \in (I_P \cup I_Q) \cap O_R$. Here, $p \parallel q \stackrel{a}{\not\longrightarrow}$ can have several reasons. We might have $p \stackrel{a}{\not\longrightarrow}$ and $a \in I_P$, or $q \stackrel{a}{\not\longrightarrow}$ and $a \in I_Q$, and in both cases $(p,q,r) \in Err_{PQR}$ due to $r \stackrel{a}{\dashrightarrow}$. Otherwise, $(p,q) \stackrel{a}{\longrightarrow} (p',q') \in E_{PQ}$; in this case, $(p,q,r) \stackrel{a}{\dashrightarrow} (p',q',r') \in E_{PQ} \times R \subseteq E_{PQR}$ by the above, implying $(p,q,r) \in E_{PQR}$ since $a \in O_{(P\otimes Q)\otimes R}$. For the induction step, consider some $(p \parallel q, r) \stackrel{\omega}{\dashrightarrow} (p' \parallel q', r') \in E_{(P\|Q)\otimes R}$; since $(p',q',r') \in E_{PQR}$ by induction hypothesis, we are done with the '$\supseteq$'-case and, thus, with establishing the desired equality.

Denoting the universal state of S by e, we now show that the state space $(P \times Q \times R) \setminus E_{PQR} \cup \{e\}$ of S coincides with the one of $(P \parallel Q) \parallel R$ (up to the name of the universal state). The states of $(P \parallel Q) \parallel R$ are:

$$\begin{aligned}
&(((P \times Q) \setminus E_{PQ} \cup \{e_{P\|Q}\}) \times R) \setminus E_{(P\|Q)\otimes R} \cup \{e\} \\
&= ((P \times Q \times R) \setminus (E_{PQ} \times R) \cup (\{e_{P\|Q}\} \times R)) \setminus E_{(P\|Q)\otimes R} \cup \{e\} \\
&= \underbrace{(P \times Q \times R) \setminus ((E_{PQ} \times R) \cup E_{(P\|Q)\otimes R})}_{=(P\times Q\times R)\setminus E_{PQR}} \cup \underbrace{(\{(e_{P\|Q}\} \times R) \setminus E_{(P\|Q)\otimes R}}_{=\emptyset} \cup \{e\})
\end{aligned}$$

For the last step, note that $(P \times Q \times R) \cap (\{e_{P\|Q}\} \times R) = \emptyset$.

Finally, we prove that the transitions of S and $(P \parallel Q) \parallel R$ are the same. For transitions to e, consider $(p \parallel q) \parallel r \stackrel{i}{\dashrightarrow} e$ for some $i \in I_{(P\|Q)\|R}$. This transition exists iff $(p \parallel q, r) \stackrel{i}{\dashrightarrow} (t, r') \in E_{(P\|Q)\otimes R}$ for some t and r'. Now, either $t = p' \parallel q'$ for some p' and q', and we have $(t,r') \in E_{(P\|Q)\otimes R} \setminus (\{e_{P\|Q}\} \times R)$; or $(p \parallel q, r) \stackrel{i}{\dashrightarrow} (e_{P\|Q}, r')$, which holds iff $(p,q) \stackrel{i}{\dashrightarrow} (p',q') \in E_{PQ}$ and either $r \stackrel{i}{\dashrightarrow} r'$ or $i \notin A_R$ and $r = r'$. This is equivalent to $(p,q,r) \stackrel{i}{\dashrightarrow} (p',q',r') \in E_{PQ} \times R$. Both cases together show: $(p \parallel q) \parallel r \stackrel{i}{\dashrightarrow} e$ iff $(p,q,r) \stackrel{i}{\dashrightarrow}_{P\otimes Q\otimes R} (p',q',r') \in E_{PQR}$ iff $(p,q,r) \stackrel{i}{\dashrightarrow}_S e$ in S.

For transitions between the states of S, which are also the states of $(P \| Q) \| R$, observe that these are exactly the transitions inherited from $(P \otimes Q) \otimes R$ minus all i-transitions from any s with $s \stackrel{i}{\dashrightarrow} e$. In $(P \parallel Q) \parallel R$, all transitions are inherited indirectly from $(P \otimes Q) \otimes R$; if $s \stackrel{i}{\dashrightarrow} e$, s clearly has no other i-transitions.

It remains for us to show that no a-transition from some state $s \in S$ is missing, if $s \overset{a}{\not\dashrightarrow} e$. Assume the contrary, namely that a transition $s = (p, q, r) \overset{a}{\dashrightarrow}_{P \otimes Q \otimes R} (p', q', r')$ of S is missing in $(P \parallel Q) \parallel R$ although $s \overset{a}{\not\dashrightarrow} e$. This can only be due to pruning; recall that $(p \parallel q) \parallel r$ and $(p' \parallel q') \parallel r'$ are states of $(P \parallel Q) \parallel R$.

If $(p, q) \overset{a}{\not\dashrightarrow}_{P \otimes Q}$, then $a \notin A_P \cup A_Q$, and the missing transition was lost when pruning $(P \parallel Q) \otimes R$, contradicting $s \overset{a}{\not\dashrightarrow} e$. Thus, $(p, q) \overset{a}{\dashrightarrow}_{P \otimes Q} (p', q')$.

If $p \parallel q \overset{a}{\not\dashrightarrow} p' \parallel q'$, then we have $p \parallel q \overset{a}{\dashrightarrow} e_{P \parallel Q}$ and $(p \parallel q, r)$ is illegal if $a \in O_R$ or $(p \parallel q) \parallel r \overset{a}{\dashrightarrow} e$, a contradiction in both cases. Thus, $(p \parallel q, r) \overset{a}{\dashrightarrow} (p' \parallel q', r')$ in $(P \parallel Q) \otimes R$. Again in this case, the transition was lost when pruning $(P \parallel Q) \otimes R$, a contradiction. $\square$

This lemma immediately implies the desired associativity:

Theorem 12. *Parallel composition is associative in the sense that, for MIAs P, Q and R, if $(P \parallel Q) \parallel R$ is defined, then $P \parallel (Q \parallel R)$ is defined and both are isomorphic, and vice versa.*

Now we proceed to show that MIA refinement is compositional wrt. parallel composition, which essentially means that $P_1 \sqsubseteq Q_1$ implies $P_1 \parallel P_2 \sqsubseteq Q_1 \parallel P_2$ for all MIAs P_1, Q_1 and P_2. The proof requires the following two lemmas:

Lemma 13 (Compatibility)**.** *For MIAs P_1, P_2 and Q_1, let E_P be the E-set of $P_1 \otimes P_2$ and E_Q be the one of $Q_1 \otimes P_2$. Further, let $p_1 \in P_1$, $p_2 \in P_2$ and $q_1 \in Q_1$ such that $p_1 \sqsubseteq q_1$. Then, $(p_1, p_2) \in E_P$ implies $(q_1, p_2) \in E_Q$.*

Proof. Let I_1/O_1 be the alphabets of P_1 and Q_1, let I_2/O_2 be the alphabets of P_2, and let I/O be the alphabets of the products. The proof is by induction on the length of a path from (p_1, p_2) to an error of $P_1 \otimes P_2$:

(Base) Let (p_1, p_2) be an error.

- Let $p_1 \overset{a}{\dashrightarrow}$ with $a \in O_1 \cap I_2$ and $p_2 \overset{a}{\not\longrightarrow}$. Then, for some q_1', we have $q_1 \overset{\varepsilon}{\Longrightarrow} q_1' \overset{a}{\dashrightarrow}$ by $p_1 \sqsubseteq q_1$; hence, $(q_1, p_2) \overset{\varepsilon}{\Longrightarrow} (q_1', p_2) \in E_Q$ and $(q_1, p_2) \in E_Q$ as well.
- Let $p_2 \overset{a}{\dashrightarrow}$ with $a \in O_2 \cap I_1$ and $p_1 \overset{a}{\not\longrightarrow}$. If $q_1 \overset{a}{\longrightarrow}$, we have a contradiction to $p_1 \sqsubseteq q_1$; otherwise, (q_1, p_2) is an error since $a \in I_1 \cap O_2$.
- If $p_1 = e_{P_1}$, then $q_1 = e_{Q_1}$ because of $p_1 \sqsubseteq q_1$, and thus $(q_1, p_2) \in E_Q$.
- Case $p_2 = e_{P_2}$ is obvious.

(Step) For a shortest path from state (p_1, p_2) to an error, consider the first transition $(p_1, p_2) \overset{\omega}{\dashrightarrow} (p_1', p_2') \in E_P$, where $\omega \in O \cup \{\tau\}$. The transition is due to either Rule (PMay1), (PMay2) or (PMay3). In all cases we find some $q_1' \in Q_1$ such that (q_1', p_2') is locally reachable from (q_1, p_2) and $p_1' \sqsubseteq q_1'$. The latter implies $(q_1', p_2') \in E_Q$ by induction hypothesis.

(PMay1) $p_1 \overset{\omega}{\dashrightarrow} p_1'$, $p_2 = p_2'$, $\omega \notin A_2$. Due to $p_1 \sqsubseteq q_1$, there is a q_1' such that $q_1 \overset{\hat{\omega}}{=\!\!\Rightarrow} q_1'$ and $p_1' \sqsubseteq q_1'$, and $(q_1, p_2) \overset{\hat{\omega}}{=\!\!\Rightarrow} (q_1', p_2)$ by applications of (PMay1). By induction hypothesis, $(q_1', p_2) \in E_Q$ and, therefore, $(q_1, p_2) \in E_Q$.

(PMay2) $p_1 = p_1'$, $p_2 \overset{\omega}{\dashrightarrow} p_2'$ and $\omega \notin A_1$. Using (PMay2) we obtain $(q_1, p_2) \overset{\omega}{\dashrightarrow} (q_1, p_2')$, so that $(q_1, p_2') \in E_Q$ by induction hypothesis. Hence, $(q_1, p_2) \in E_Q$, too.

(PMay3) $\omega = o$, $p_1 \overset{o}{\dashrightarrow} p_1'$ and $p_2 \overset{o}{\dashrightarrow} p_2'$ with $o \in A_1 \cap A_2$. Note that o is an output for the product and one of its components, but an input for the other. By $p_1 \sqsubseteq q_1$ we have $q_1 \overset{\varepsilon}{=\!\!\Rightarrow} q_1'' \overset{o}{\dashrightarrow} q_1''' \overset{\varepsilon}{=\!\!\Rightarrow} q_1'$ for some q_1', q_1'', q_1''' with $p_1' \sqsubseteq q_1'$. (Note, that in case $o \in I_1$ we have $q_1 = q_1''$.) Therefore, we get $(q_1, p_2) \overset{\varepsilon}{=\!\!\Rightarrow} (q_1'', p_2) \overset{o}{\dashrightarrow} (q_1''', p_2') \overset{\varepsilon}{=\!\!\Rightarrow} (q_1', p_2')$ via (PMay1) and (PMay3). By induction hypothesis, $(q_1', p_2') \in E_Q$ and, hence, $(q_1, p_2) \in E_Q$, too. □

The next lemma generalises the synchronisation according to Rule (PMust3) to weak transitions:

Lemma 14 (Weak Must-Transitions)**.** *Let P, Q be composable MIAs.*

1. *For $\alpha \notin A_Q$, $p \overset{\hat{\alpha}}{\Longrightarrow} P'$ and $q \in Q$ implies $(p, q) \overset{\hat{\alpha}}{\Longrightarrow} P' \times \{q\}$ in $P \otimes Q$.*
2. *If $p \overset{a}{\Longrightarrow} P'$ (or $p \overset{a}{\longrightarrow}\overset{\varepsilon}{\Longrightarrow} P'$) and $q \overset{a}{\longrightarrow} Q'$ for some $a \in A_P \cap A_Q$, then $(p, q) \overset{a}{\Longrightarrow} P' \times Q'$ (or $(p, q) \overset{a}{\longrightarrow}\overset{\varepsilon}{\Longrightarrow} P' \times Q'$) in $P \otimes Q$.*

Proof. Claim 1: Clearly, the mapping $P \to P \times \{q\} : p \mapsto (p, q)$ is an isomorphism if we only consider must-transitions labelled with the given α or τ and states in $P \times \{q\}$ in $P \otimes Q$.

Claim 2: By induction on the definition of $p \overset{a}{\Longrightarrow} P'$. In Case 2 of Def. 2, we have $p \overset{\tau}{\longrightarrow} \bar{P}$ and a suitable $\bar{p} \overset{a}{\Longrightarrow} P_{\bar{p}}$ for each $\bar{p} \in \bar{P}$, such that $P' = \bigcup_{\bar{p} \in \bar{P}} P_{\bar{p}}$. Then, $(p, q) \overset{\tau}{\longrightarrow} \bar{P} \times \{q\}$ due to (PMust1), and $(\bar{p}, q) \overset{a}{\Longrightarrow} P_{\bar{p}} \times Q'$ by induction hypothesis; this yields $(p, q) \overset{a}{\Longrightarrow} P' \times Q'$ due to Def. 2.2. In Case 3 (the only one for the variant concerning $\overset{a}{\longrightarrow}\overset{\varepsilon}{\Longrightarrow}$), we have $p \overset{a}{\longrightarrow} \bar{P}$ and a suitable $\bar{p} \overset{\varepsilon}{\Longrightarrow} P_{\bar{p}}$ for each $\bar{p} \in \bar{P}$ such that $P' = \bigcup_{\bar{p} \in \bar{P}} P_{\bar{p}}$. Then, $(p, q) \overset{a}{\longrightarrow} \bar{P} \times Q'$ by (PMust3) and, for each $(\bar{p}, q') \in \bar{P} \times Q'$, we get $(\bar{p}, q') \overset{\varepsilon}{\Longrightarrow} P_{\bar{p}} \times \{q'\}$ by Claim 1, hence $\bar{P} \times Q' \overset{\varepsilon}{\Longrightarrow} P' \times Q'$. By Def. 2.3 we obtain $(p, q) \overset{a}{\Longrightarrow} P' \times Q'$. □

Theorem 15 (Compositionality of Parallel Composition)**.** *Let P_1, P_2 and Q_1 be MIAs and $P_1 \sqsubseteq Q_1$. Assume that Q_1 and P_2 are composable, then:*

1. *P_1 and P_2 are composable.*
2. *$P_1 \parallel P_2 \sqsubseteq Q_1 \parallel P_2$, and $P_1 \parallel P_2$ is compatible if $Q_1 \parallel P_2$ is.*

Proof. Part 1 is trivial. Regarding Part 2, the second claim is immediate from the first claim and Lem. 13. We denote the universal state of $P_1 \| P_2$ and $Q_1 \| P_2$ by e_P and e_Q, resp. E_P stands for the E-set of $P_1 \otimes P_2$ and E_Q for the one of $Q_1 \otimes P_2$, as in Lem. 13. To establish the first claim of Part 2, we prove that

$$\mathcal{R} =_{\mathrm{df}} \{(p_1 \| p_2,\ q_1 \| p_2) \mid p_1 \sqsubseteq q_1\} \cup ((P_1 \| P_2) \times \{e_Q\})$$

is a MIA-refinement relation by checking the conditions of Def. 4. Then, we are done since $p_{01} \sqsubseteq q_{01}$ due to $P_1 \sqsubseteq Q_1$ and, therefore, $(p_{01} \| p_{02}, q_{01} \| p_{02}) \in \mathcal{R}$. For the second subset, the check is trivial; so consider some $(p_1 \| p_2, q_1 \| p_2) \in \mathcal{R}$:

(i) Obvious, since $p_1 \| p_2 \neq e_P$.

(ii) Let $q_1 \| p_2 \xrightarrow{i} \bar{Q}$ due to either Rule (PMust1), (PMust2) or (PMust3). Note that $(q_1, p_2) \xrightarrow{i} \bar{Q}$ in $Q_1 \otimes P_2$ as well. If any state pair in $\bar{Q}$ was illegal, the transition would have been removed by pruning.

(PMust1) $q_1 \xrightarrow{i} Q'_1$ and $\bar{Q} = Q'_1 \times \{p_2\}$. Then, by $p_1 \sqsubseteq q_1$, there is a $P'_1 \subseteq P_1$ such that $p_1 \xrightarrow{i} \overset{\varepsilon}{\Longrightarrow}_{P_1} P'_1$ and $\forall p'_1 \in P'_1\, \exists q'_1 \in Q'_1.\, p'_1 \sqsubseteq q'_1$. Now, $(p_1, p_2) \xrightarrow{i} \overset{\varepsilon}{\Longrightarrow} P'_1 \times \{p_2\}$ by repeated application of Rule (PMust1) and since $i \notin A_2$. For every $(p'_1, p_2) \in P'_1 \times \{p_2\}$, we have a suitable $(q'_1, p_2) \in Q'_1 \times \{p_2\}$; moreover, $(p'_1, p_2) \notin E_P$ since $(q'_1, p_2) \notin E_Q$ and by Lem. 13. Thus, we have $(p'_1 \| p_2, q'_1 \| p_2) \in \mathcal{R}$.

It remains for us to show that $(p_1, p_2) \xrightarrow{i} \overset{\varepsilon}{\Longrightarrow} P'_1 \times \{p_2\}$ also exists in $P_1 \| P_2$, i.e., that no state (p''_1, p_2) along this weak transition is pruned. More generally, let us consider any $\bar{p}_1$ and p''_1 with $p_1 \overset{i}{\dashrightarrow} \bar{p}_1 \overset{\varepsilon}{\Longrightarrow} p''_1$, implying $(p_1, p_2) \overset{i}{\dashrightarrow} (\bar{p}_1, p_2) \overset{\varepsilon}{\Longrightarrow} (p''_1, p_2)$. Because of $p_1 \overset{i}{\dashrightarrow} \bar{p}_1$ and $p_1 \sqsubseteq q_1$, there must be some $\bar{q}_1$ with $q_1 \overset{i}{\dashrightarrow} \overset{\varepsilon}{\Longrightarrow} \bar{q}_1$ which implies $(q_1, p_2) \overset{i}{\dashrightarrow} \overset{\varepsilon}{\Longrightarrow} (\bar{q}_1, p_2)$, and $\bar{p}_1 \sqsubseteq \bar{q}_1$. If $(\bar{q}_1, p_2) \in E_Q$, then all outgoing i-transitions from $q_1 \| p_2$ would have been pruned, contradicting our assumptions. Thus, and by Lem. 13, $(\bar{p}_1, p_2) \notin E_P$, which means that $(p''_1, p_2) \notin E_P$, too.

(PMust2) $p_2 \xrightarrow{i} P'_2$ and $\bar{Q} = \{q_1\} \times P'_2$. Then, $(p_1, p_2) \xrightarrow{i} \bar{P} = \{p_1\} \times P'_2$ according to (PMust2) and since $i \notin A_1$. For $(p_1, p'_2) \in \bar{P}$, we get $(p_1, p'_2) \notin E_P$ because $(q_1, p'_2) \notin E_Q$ and due to Lem. 13. Thus, $p_1 \| p_2 \xrightarrow{i} \bar{P}$ and, for every $p_1 \| p'_2 \in \bar{P}$, we have $q_1 \| p'_2 \in \bar{Q}$ with $(p_1 \| p'_2, q_1 \| p'_2) \in \mathcal{R}$.

(PMust3) $q_1 \xrightarrow{i} Q'_1$, $p_2 \xrightarrow{i} P'_2$ and $\bar{Q} = Q'_1 \times P'_2$. (Note that $i \in I_1 \cap I_2$.) Then, by $p_1 \sqsubseteq q_1$, there is a set $P'_1 \subseteq P_1$ such that $p_1 \xrightarrow{i} \overset{\varepsilon}{\Longrightarrow} P'_1$ and $\forall p'_1 \in P'_1\, \exists q'_1 \in Q'_1.\, p'_1 \sqsubseteq q'_1$. By Lem. 14 we get $(p_1, p_2) \xrightarrow{i} \overset{\varepsilon}{\Longrightarrow} P'_1 \times P'_2$. Similarly to Case (PMust1), we have to show that $(p_1, p_2) \xrightarrow{i} \overset{\varepsilon}{\Longrightarrow} P'_1 \times P'_2$ also exists in $P_1 \| P_2$, i.e., no state (p''_1, p'_2) along this weak transition is pruned. More generally, let us consider any $\bar{p}_1$ and p_1'' with $p_1 \overset{i}{\dashrightarrow} \bar{p}_1 \overset{\varepsilon}{\Longrightarrow} p_1''$ and some p_2' with $p_2 \overset{i}{\dashrightarrow} p_2'$, implying $(p_1, p_2) \overset{i}{\dashrightarrow} (\bar{p}_1, p_2') \overset{\varepsilon}{\Longrightarrow} (p_1'', p_2')$. Because of $p_1 \overset{i}{\dashrightarrow} \bar{p}_1$ and $p_1 \sqsubseteq q_1$, there must be some $\bar{q}_1$ with $q_1 \overset{i}{\dashrightarrow}\overset{\varepsilon}{\Longrightarrow} \bar{q}_1$, which implies $(q_1, p_2) \overset{i}{\dashrightarrow}\overset{\varepsilon}{\Longrightarrow} (\bar{q}_1, p_2')$, and $\bar{p}_1 \sqsubseteq \bar{q}_1$. If $(\bar{q}_1, p_2') \in E_Q$, then all outgoing i-transitions from $q_1 \| p_2$ would have been pruned, contradicting our assumptions. Therefore, and by Lem. 13, $(\bar{p}_1, p_2') \notin E_P$, which means that $(p_1'', p_2') \notin E_P$, too.

(iii) Let $q_1 \parallel p_2 \overset{\omega}{\longrightarrow} \bar{Q}$ due to either (PMust1), (PMust2) or (PMust3). Again the transition and the states exist in $Q_1 \otimes P_2$, too, as argued above.

(PMust1) $q_1 \overset{\omega}{\longrightarrow} Q_1'$, $\omega \notin A_2$ and $\bar{Q} = Q_1' \times \{p_2\}$. Then, by $p_1 \sqsubseteq q_1$, there exists $P_1' \subseteq P_1$ such that $p_1 \overset{\hat{\omega}}{\Longrightarrow} P_1'$ and $\forall p_1' {\in} P_1' \, \exists q_1' {\in} Q_1'.\, p_1' \sqsubseteq q_1'$. Now, $(p_1, p_2) \overset{\hat{\omega}}{\Longrightarrow} P_1' \times \{p_2\}$ according to (PMust1) and since $\omega \notin A_2$. Because p_1 and p_2 are compatible, this also holds for all pairs along this weak transition by the definition of E_P. For $p_1' \in P_1'$ we have a suitable $q_1' \in Q_1'$ such that, for the arbitrary $p_1' \parallel p_2$, we may also infer $(p_1' \parallel p_2, q_1' \parallel q_2) \in \mathcal{R}$.

(PMust2) $p_2 \overset{\omega}{\longrightarrow}_{P_2} P_2'$, $\omega \notin A_1$ and $\bar{Q} = \{q_1\} \times P_2'$. In this case we obtain that $(p_1, p_2) \overset{\omega}{\longrightarrow} \bar{P} = \{p_1\} \times P_2'$ by (PMust2) and $\omega \notin A_1$. For $(p_1, p_2') \in \bar{P}$ we get $(p_1, p_2') \notin E_P$ since $(q_1, p_2') \notin E_Q$ and due to Lem. 13. Thus, $p_1 \parallel p_2 \overset{\omega}{\longrightarrow} \bar{P}$ and therefore also $p_1 \parallel p_2 \overset{\hat{\omega}}{\Longrightarrow} \bar{P}$. For $(p_1, p_2') \in \bar{P}$ we also have $(p_1 \parallel p_2', q_1 \parallel p_2') \in \mathcal{R}$.

(PMust3) $\omega = o$, $q_1 \overset{o}{\longrightarrow} Q_1'$, $p_2 \overset{o}{\longrightarrow} P_2'$ for some action $o \in (O_1 \cap I_2) \cup (I_1 \cap O_2)$, and $\bar{Q} = Q_1' \times P_2'$. By $p_1 \sqsubseteq q_1$, there exists some $P_1' \subseteq P_1$ with $p_1 \overset{o}{\Longrightarrow} P_1'$ (possibly $p_1 \overset{o}{\longrightarrow}\overset{\varepsilon}{\Longrightarrow} P_1'$, if $o \in I_1$) such that $\forall p_1' {\in} P_1'$ $\exists q_1' {\in} Q_1'.\ p_1' \sqsubseteq q_1'$. Now, $(p_1, p_2) \overset{o}{\Longrightarrow} R \subseteq P_1' \times P_2'$ by Lem. 14 and, as in Case (PMust1) above, all pairs along this weak transition are compatible. Hence, $p_1 \parallel p_2 \overset{o}{\Longrightarrow} R$ and, for all $p_1' \parallel p_2' \in R$, we have some $q' \in Q'$ such that $(p_1' \parallel p_2', q_1' \parallel p_2') \in \mathcal{R}$.

(iv) First, consider $p_1 \| p_2 \overset{i}{\dashrightarrow} e_P$ due to pruning, i.e., $(p_1, p_2) \overset{i}{\dashrightarrow} (p_1', p_2') \in E_P$.

(PMay1) $p_1 \overset{i}{\dashrightarrow}_{P_1} p_1'$, $i \notin A_2$ and $p_2' = p_2$. By $p_1 \sqsubseteq q_1$, we have $q_1 \overset{i}{\dashrightarrow} q_1'' \overset{\varepsilon}{\Longrightarrow} q_1'$ for some q_1', q_1'' such that $p_1' \sqsubseteq q_1'$. Hence, $(q_1, p_2) \overset{i}{\dashrightarrow} (q_1'', p_2) \overset{\varepsilon}{\Longrightarrow} (q_1', p_2)$ by repeated application of (PMay1) and since $i \notin A_2$. By Lem. 13 we get $(q_1', p_2) \in E_Q$ and thus $(q_1'', p_2) \in E_Q$. Therefore, $q_1 \parallel p_2 \overset{i}{\dashrightarrow} e_Q$ by pruning.

(PMay2) $p_2 \overset{i}{\dashrightarrow} p_2'$, $i \notin A_1$ and $p_1' = p_1$. Then, $(q_1, p_2) \overset{i}{\dashrightarrow} (q_1, p_2')$ by (PMay2). By Lem. 13 we get $(q_1, p_2') \in E_Q$. Hence, $q_1 \| p_2 \overset{i}{\dashrightarrow} e_Q$ by pruning.

(PMay3) $p_1 \stackrel{i}{\dashrightarrow} p_1'$ and $p_2 \stackrel{i}{\dashrightarrow} p_2'$ for some action $i \in I_1 \cap I_2$. Due to $p_1 \sqsubseteq q_1$, we get $q_1 \stackrel{i}{\dashrightarrow} q_1'' \stackrel{\varepsilon}{\Longrightarrow} q_1'$ for some q_1', q_1'' such that $p_1' \sqsubseteq q_1'$. Hence, $(q_1, p_2) \stackrel{i}{\dashrightarrow} (q_1'', p_2') \stackrel{\varepsilon}{\Longrightarrow} (q_1', p_2')$ by Rules (PMay1) and (PMay3). Lem. 13 yields $(q_1', p_2') \in E_Q$, and thus $(q_1'', p_2') \in E_Q$ as well. Therefore, $q_1 \parallel p_2 \stackrel{i}{\dashrightarrow} e_Q$ by pruning.

Second, we consider $p_1 \parallel p_2 \stackrel{i}{\dashrightarrow} p_1' \parallel p_2'$, due to one of the Rules (PMay1), (PMay2) or (PMay3).

(PMay1) $p_1 \stackrel{i}{\dashrightarrow} p_1'$, $i \notin A_2$ and $p_2' = p_2$. By $p_1 \sqsubseteq q_1$, we have $q_1 \stackrel{i}{\dashrightarrow}\stackrel{\varepsilon}{\Longrightarrow} q_1'$ for some q_1' such that $p_1' \sqsubseteq q_1'$. Hence, $(q_1, p_2) \stackrel{i}{\dashrightarrow}\stackrel{\varepsilon}{\Longrightarrow} (q_1', p_2)$ by repeated application of (PMay1) and since $i \notin A_2$. If any state along this weak transition is in E_Q, then we get $q_1 \parallel p_2 \stackrel{i}{\dashrightarrow} e_Q$ and $(p_1' \parallel p_2', e_Q) \in \mathcal{R}$. Otherwise, $q_1 \parallel p_2 \stackrel{i}{\dashrightarrow}\stackrel{\varepsilon}{\Longrightarrow} q_1' \parallel p_2$ with $(p_1' \parallel p_2, q_1' \parallel p_2) \in \mathcal{R}$.

(PMay2) $p_2 \stackrel{i}{\dashrightarrow} p_2'$, $i \notin A_1$ and $p_1' = p_1$. Then, $(q_1, p_2) \stackrel{i}{\dashrightarrow} (q_1, p_2')$ by (PMay2). If the latter state (q_1, p_2') is in E_Q, then $q_1 \parallel p_2 \stackrel{i}{\dashrightarrow} e_Q$ and are done. Otherwise we have $(p_1 \parallel p_2', q_1 \parallel p_2') \in \mathcal{R}$.

(PMay3) $p_1 \stackrel{i}{\dashrightarrow} p_1'$ and $p_2 \stackrel{i}{\dashrightarrow} p_2'$ for some action $i \in I_1 \cap I_2$: Due to $p_1 \sqsubseteq q_1$, we get $q_1 \stackrel{i}{\dashrightarrow} q_1'' \stackrel{\varepsilon}{\Longrightarrow} q_1'$ for some $q_1', q_1'' \in Q$ such that $p_1' \sqsubseteq q_1'$. Now, we obtain $(q_1, p_2) \stackrel{i}{\dashrightarrow} (q_1'', p_2') \stackrel{\varepsilon}{\Longrightarrow} (q_1', p_2')$ by (PMay1) and (PMay3). If any state along $(q_1'', p_2') \stackrel{\varepsilon}{\Longrightarrow} (q_1', p_2')$ is in E_Q, then we get $(q_1, p_2) \stackrel{i}{\dashrightarrow} e_Q$ and $(p_1' \parallel p_2', e_Q) \in \mathcal{R}$. Otherwise, we again have $(p_1' \parallel p_2', q_1' \parallel p_2') \in \mathcal{R}$.

(v) Let $p_1 \parallel p_2 \stackrel{\omega}{\dashrightarrow} p_1' \parallel p_2'$, due to one of the Rules (PMay1) through (PMay3).

(PMay1) $p_1 \stackrel{\omega}{\dashrightarrow} p_1'$, $\omega \notin A_2$ and $p_2' = p_2$. By $p_1 \sqsubseteq q_1$, we have $q_1 \stackrel{\hat{\omega}}{\Longrightarrow} q_1'$ for some q_1' such that $p_1' \sqsubseteq q_1'$. Hence, $(q_1, p_2) \stackrel{\hat{\omega}}{\Longrightarrow} (q_1', p_2)$ by repeated application of (PMay1) and since $\omega \notin A_2$. If any state along this weak transition was in E_Q, then also $(q_1, p_2) \in E_Q$, which contradicts $(p_1 \parallel p_2, q_1 \parallel p_2) \in \mathcal{R}$. Thus, $q_1 \parallel p_2 \stackrel{\hat{\omega}}{\Longrightarrow} q_1' \parallel p_2$ with $(p_1' \parallel p_2, q_1' \parallel p_2) \in \mathcal{R}$.

(PMay2) $p_2 \stackrel{\omega}{\dashrightarrow} p_2'$, $\omega \notin A_1$ and $p_1' = p_1$. Then, $(q_1, p_2) \stackrel{\omega}{\dashrightarrow} (q_1, p_2')$ by (PMay2) and due to $p_1 \sqsubseteq q_1$. If the latter state (q_1, p_2') were in E_Q, then also the former state (q_1, p_2) would be in E_Q. Thus, we have $q_1 \| p_2 \stackrel{\omega}{\dashrightarrow} q_1 \| p_2'$ and $(p_1 \| p_2', q_1 \| p_2') \in \mathcal{R}$.

(PMay3) $\omega = o$, $p_1 \stackrel{o}{\dashrightarrow} p_1'$ and $p_2 \stackrel{o}{\dashrightarrow} p_2'$ for some action $o \in (O_1 \cap I_2) \cup (I_1 \cap O_2)$. Due to $p_1 \sqsubseteq q_1$, we get $q_1 \stackrel{\varepsilon}{\Longrightarrow} q_1'' \stackrel{o}{\dashrightarrow} q_1''' \stackrel{\varepsilon}{\Longrightarrow} q_1'$ (or

$q_1 \overset{o}{\dashrightarrow} q_1''' \overset{\varepsilon}{\Longrightarrow} q_1'$ if $o \in I_1$) for some $q_1', q_1'', q_1''' \in Q$ such that $p_1' \sqsubseteq q_1'$. Now, we obtain $(q_1, p_2) \overset{\varepsilon}{\Longrightarrow} (q_1'', p_2) \overset{o}{\dashrightarrow} (q_1''', p_2') \overset{\varepsilon}{\Longrightarrow} (q_1', p_2')$ (or $(q_1, p_2) \overset{o}{\dashrightarrow} (q_1''', p_2') \overset{\varepsilon}{\Longrightarrow} (q_1', p_2')$) by (PMay1) and (PMay3). Hence, $q_1 \parallel p_2 \overset{o}{\Longrightarrow} q_1' \parallel p_2'$ and $(p_1' \parallel p_2', q_1' \parallel p_2') \in \mathcal{R}$, as in Case (PMay1) above. □

We close this subsection on parallel composition with a discussion of legal environments as introduced for IA in [4]. Intuitively, a legal (or helpful) environment for a composition $P \otimes Q$ is a MIA V that prevents $P \otimes Q$ from running into an error. In the final application, the parallel composition is embedded in such a legal environment, which may for example represent a user. In [4], it is shown that two systems are compatible (i.e., their parallel composition is defined) if and only if there is a legal environment for them. This justifies to some degree the pruning in Def. 8: the parallel composition of two IAs is undefined due to the initial state being removed by pruning if and only if no environment can use them without producing errors. Correspondingly, two MIAs are incompatible if the initial state of the parallel composition is set to e due to pruning; this special state indicates that the composition is not defined properly.

Definition 16 (Legal Environment)**.** *A* legal environment *for MIAs P and Q is a MIA V with:*

1. *V is composable with $P \otimes Q$,*
2. *$I_{(P \otimes Q) \otimes V} = \emptyset$,*
3. *The reachable states of $(P \otimes Q) \otimes V$ contain neither new nor inherited errors in the sense of Lem. 11.*

Note that, since $(P \otimes Q) \otimes V$ only has locally controlled actions, all reachable errors are locally reachable. Usually, in frameworks with binary communication, an environment is defined to have the outputs of $P \otimes Q$ as inputs and vice versa; due to hiding of synchronized actions, their composition is closed. Here, such a signature results in the product only having output and internal actions, which is natural for multicasting such as ours. One can then close the system with hiding all outputs. Similarly, in Cond. 2 of Def. 16 we require that composition with an environment results in a system without inputs.

Proposition 17. *MIAs P and Q are compatible if and only if there exists a legal environment for them.*

Proof. '⇒': If P and Q are compatible, then $P \otimes Q$ has no locally reachable errors. Composing it with a MIA V that accepts all inputs (via must-loops at the initial state) but provides no outputs, yields only those states that are locally reachable from (p_0, q_0). Thus, V is a legal environment for P and Q.

'⇐': Assume, towards a contradiction, that P and Q are incompatible, i.e., $P \otimes Q$ has a locally reachable error (p, q). Then, for any MIA V, $(P \otimes Q) \otimes V$ either has a reachable state $((p, q), v)$ resulting in an inherited error, or there is a first output transition on the path to (p, q) that V prevents by not providing the corresponding input transition. This results in a locally reachable new error in $(P \otimes Q) \otimes V$. Either way, V is no legal environment. $\square$

We can do better than this to justify pruning. The next proposition shows that pruning only removes behaviour from $P \otimes Q$ that is never reached in any legal environment. In other words, pruning does not change the behaviour when the composition is used properly, i.e., in a legal environment. Note that our result is actually more general, since it holds for all MIAs V that satisfy Conds. 1 and 3 of Def. 16.

Proposition 18. *For MIAs P and Q and a corresponding legal environment V, we have $(P \otimes Q) \otimes V = (P \parallel Q) \otimes V$ (up to the names of the respective universal states).*

Proof. Due to Def. 16, the pruning in Lem. 11 does not change $(P \otimes Q) \otimes V$ and, in the latter, the universal state is unreachable. Furthermore, it equals $(P \parallel Q) \parallel V$ (up to the names of universal states). Since the universal state is unreachable in $(P \parallel Q) \parallel V$, pruning of $(P \parallel Q) \otimes V$ left the MIA unchanged and the claim follows. $\square$

3.2. Universal States in Input/Output Approaches

States that, like our universal state e, represent arbitrary behaviour and have only input transitions as ingoing transitions date back at least to the thesis of Dill [25]. His work is focussed on a trace-based semantics, consisting of a set of ordinary traces and a set of so-called failure traces. The latter deal with behaviour resulting from communication mismatches. LTS-representations of the semantics have a special state that has arbitrary behaviour due to looping transitions for all actions. This state *completes* the LTS by making the other states input-enabled, and it is not an ordinary state since it is the only one representing the failure traces. The notion of input-enabledness is purely syntactic: a state s is input-enabled if it has an outgoing i-transition (must or may in case of modalities), for each input i. Considering an LTS in [25], an i-transition from s to the special state indicates that s cannot safely receive this input. This is just the same as a missing input in IA, and the LTS really *is* an IA. The representation based on a special state is just more convenient for a trace-based semantics expressing that, as in standard IA, a missing input can always be added in a refinement step.

A similar completion can be found in a process-algebraic setting in [26], where it is called *demonic*: if a process p does not have an i-transition according to the standard operational rules, then there are additional rules that give p an i-transition to a process having arbitrary behaviour, essentially due to loops. The latter process is an ordinary process, and communication mismatches are not considered. A variant of demonic completion is also used in [27] to achieve compositionality for parallel composition in the *ioco*-approach to conformance testing; this approach also disregards communication mismatches. The suggestion is to apply the completion to the specification first, each time the ioco-implementation relation is checked. That the completion uses ordinary states makes sense only because ioco does not support stepwise refinement and assumes that implementations are always input-enabled. Applying the suggested solution in IA would force each refinement to be input-enabled, violating the very idea of IA.

This problem with ordinary universal states in an IA-approach can be fixed as in [6, 24], where universal states are called *error states*. The *semantics* and the special treatment of error states in these papers is similar to the one in [25], but error states do not necessarily complete an IA and do not need loops. They arise in case of a communication mismatch in a parallel composition, just as in the present paper. The problem with ordinary universal states vanishes when modalities are added, since input-transitions to the universal state and the loops at this state can be declared to be of may-modality. Completion with this idea is used in [8] when translating IA with their refinement relation to MTSs. There, an input may-transition always expresses that the resp. input is allowed in a refinement, but at present the input cannot be received safely.

As already discussed above, an ordinary state tt with may-loops is inserted during parallel composition in [10] as target for input transitions that have been cut due to pruning. This way, a precongruence is achieved, and it works fine for refinement that tt is regarded as an ordinary state. However, parallel composition is not associative this way; to avoid this problem, we insert e during parallel composition *and* give it a special treatment in refinement. It is important to note that we do *not* perform completion, i.e., for some ordinary state, an input can also be forbidden in all refinements, in accordance with the MTS-view.

3.3. Hiding and Restriction

We now introduce operators for scoping actions, namely hiding [13] and restriction [14], as is usual in process algebra. In our setting, outputs are under the control of the system; when disconnected, they are still performed but the signal is no longer sent to the outside, i.e., the action is internal. In contrast, inputs are only performed because of an outside stimulus. Disconnecting an input rather blocks it and, therefore, we introduce a restriction operator for inputs. The same idea is used in the IA-setting of [28], but hiding and restriction are combined into one operation.

Definition 19 (Hiding)**.** *Given a MIA $P = (P, I, O, \longrightarrow_P, \dashrightarrow_P, p_0, e)$ and a set L of actions with $L \cap I = \emptyset$. Then, P* hiding *L is the MIA $P/L =_{df} (P, I, O \setminus L, \longrightarrow_{P/L}, \dashrightarrow_{P/L}, p_0, e)$, where all transition labels $o \in L$ are replaced by τ.*

Definition 20 (Restriction)**.** *Given a MIA $P = (P, I, O, \longrightarrow_P, \dashrightarrow_P, p_0, e)$ and a set L of actions such that $L \cap O = \emptyset$. Then,* restricting *L in P yields the MIA $P \setminus L =_{df} (P, I \setminus L, O, \longrightarrow_{P\setminus L}, \dashrightarrow_{P\setminus L}, p_0, e_P)$, where all transitions with a label contained in L are deleted.*

Observe that hiding and restriction yield well-defined MIAs; in particular, the sink condition is preserved by hiding since $L \cap I = \emptyset$.

Lemma 21 (Weak Must-Transitions under Hiding)**.** *Let P be a MIA, $L \cap I = \emptyset$ and $o \in L \cap O$. If $p \overset{o}{\Longrightarrow}_P P'$, then $p \overset{\varepsilon}{\Longrightarrow}_{P/L} P'$.*

Proof. By induction on the definition of $p \overset{o}{\Longrightarrow}_P P'$. If $p \overset{o}{\Longrightarrow}_P P'$ is due to Def. 2.3, then the claim is obvious. Otherwise, $p \overset{o}{\Longrightarrow}_P P'$ is due to some $p \overset{\tau}{\longrightarrow}_P \bar{P}$ and $\bar{P} \overset{o}{\Longrightarrow}_P P'$ according to Def. 2.2. By induction hypothesis, we have $\bar{p} \overset{\varepsilon}{\Longrightarrow}_{P/L} P_{\bar{p}}$ for each $\bar{p} \in \bar{P}$ and $P' = \bigcup_{\bar{p} \in \bar{P}} P_{\bar{p}}$. By Def. 2.2, we obtain $p \overset{\varepsilon}{\Longrightarrow}_{P/L} P'$. □

As desired, MIA-refinement is a precongruence wrt. hiding and restriction:

Proposition 22. *Let P, Q be MIAs with $P \sqsubseteq Q$.*

1. *$P/L \sqsubseteq Q/L$ for any set L of actions with $L \cap I = \emptyset$.*
2. *$P \setminus L \sqsubseteq Q \setminus L$ for any set L of actions with $L \cap O = \emptyset$.*

Proof. Since $P \sqsubseteq Q$, there is a MIA-refinement relation $\mathcal{R}$ with $(p, q) \in \mathcal{R}$. We show that $\mathcal{R}$ is also a MIA-refinement relation for $P/L \sqsubseteq Q/L$ and $P \setminus L \sqsubseteq Q \setminus L$. The only interesting case concerns hiding and Def. 4(iii), i.e., $q \overset{\tau}{\longrightarrow}_{Q/L} Q'$ due to $q \overset{o}{\longrightarrow}_Q Q'$ for $o \in O \cap L$. The latter is matched by a transition $p \overset{o}{\Longrightarrow}_P P'$ with $\forall p' \in P' \, \exists q' \in Q'. \, (p', q') \in \mathcal{R}$. By Lem. 21, this yields $p \overset{\varepsilon}{\Longrightarrow}_{P/L} P'$. □

3.4. Parallel Composition with Hiding

We now turn our attention to parallel composition with immediate hiding on synchronised actions, thereby enforcing binary communication. This parallel composition is used by de Alfaro and Henzinger for Interface Automata (IA) in [23, 4]. We show here that the standard IA parallel composition can be expressed via our multicast parallel composition and hiding.

Definition 23 (Parallel Product and Composition with Hiding)**.** *MIAs P_1 and P_2 are* H-composable *if $O_1 \cap O_2 = \emptyset = I_1 \cap I_2$. We then define the* product with hiding *in the same way as the parallel product in Def. 7, except for $O =_{df} (O_1 \cup O_2) \setminus (I_1 \cup I_2)$ and a change of Rules (PMust3) and (PMay3):*

(PMust3')	$(p_1, p_2) \overset{\tau}{\longrightarrow} P_1' \times P_2'$	*if*	$p_1 \overset{a}{\longrightarrow} P_1'$ *and* $p_2 \overset{a}{\longrightarrow} P_2'$ *for some* a,
(PMay3')	$(p_1, p_2) \overset{\tau}{\dashrightarrow} (p_1', p_2')$	*if*	$p_1 \overset{a}{\dashrightarrow} p_1'$ *and* $p_2 \overset{a}{\dashrightarrow} p_2'$ *for some* a.

From this parallel product with hiding, we get the parallel composition with hiding *$P_1 \mid P_2$ by the same pruning procedure as in Def. 8.*

It can easily be seen that the parallel product with hiding can be expressed by our parallel product without hiding and the hiding operator. Pruning does not change this, since it treats outputs and internal actions equally.

Proposition 24. *Let P_1, P_2 be H-composable MIAs and $S = A_1 \cap A_2$ be the set of synchronising actions. Then, $P_1 \mid P_2 = (P_1 \parallel P_2)/S$.*

Associativity is a natural property of parallel composition, so one would expect that $(P \mid Q) \mid R \;=\; P \mid (Q \mid R)$ for some suitable equivalence $=$ (e.g., equality up to isomorphism) provided that one side is defined. This law looks much less natural if we rewrite it according to Prop. 24; it is wrong in the version of $\mid$ in [23]. Here, associativity can be proved from Thm. 12 and the following proposition.

Proposition 25. *For composable MIAs P and Q we have the following laws, where $=$ means that the respective MIAs are identical (up to the naming of the resp. universal states in Part (iii)).*

(i) $P/L = P$ *if* $A_P \cap L = \emptyset$.

(ii) $P/L/L' = P/(L \cup L')$ *if* $L \cap I_P = L' \cap I_P = \emptyset$.

(iii) $(P \parallel Q)/L = (P/L) \parallel (Q/L)$ *if* $A_P \cap A_Q \cap L = \emptyset$.

Proof. Parts (i) and (ii) are straightforward. We thus focus on proving Part (iii). $P \otimes Q$ and $P/L \otimes Q/L$ are the same due to the condition $A_P \cap A_Q \cap L$, except that transition labels $o \in L$ in the former are replaced by τ in the latter; observe that (PMust3) and (PMay3) are never applicable to $o \in L$ by assumption, and the other rules work for $o \in L$ and τ in the same way. Also by assumption, the same states are considered as errors in both products. As a consequence and since pruning makes no difference between output- and τ-transitions, it deletes the same states in both systems and the same input transitions get redirected to the respective universal states of the parallel compositions. Finally, applying hiding to $P \parallel Q$ for the first system makes the MIAs identical. $\square$

Using this proposition we may now prove the associativity of $\mid$.

Proposition 26. *Parallel composition with hiding is associative in the sense, that for pairwise H-composable MIAs P, Q and R, if $(P \mid Q) \mid R$ is defined, then $P \mid (Q \mid R)$ is defined as well and both are isomorphic, and vice versa.*

Proof. Let P, Q, R be pairwise H-composable MIAs. We set $S_{PQ} =_{\mathrm{df}} A_P \cap A_Q$, $A_{PQ} =_{\mathrm{df}} (A_P \cup A_Q) \setminus S_{PQ}$, etc. and let $S_{PQR} =_{\mathrm{df}} S_{PQ} \cup S_{PR} \cup S_{QR}$. Note that $(*)$ $S_{PQ} \cap A_R = \emptyset$ since, otherwise A_R would contain an action that is an input in one of P and Q and an output in the other, contradicting H-composability of R with one of the other MIAs. Furthermore, $(**)$ $S_{PQ} \cup (A_{PQ} \cap A_R) = S_{PQ} \cup (((A_P \cup A_Q)/S_{PQ}) \cap A_R) \stackrel{(*)}{=} S_{PQ} \cup (((A_P \cup A_Q) \cap A_R)/S_{PQ}) = S_{PQ} \cup ((A_P \cup A_Q) \cap A_R) = S_{PQ} \cup (A_P \cap A_R) \cup (A_Q \cap A_R) = S_{PQR}$. We now obtain:

$$\begin{aligned}
(P \mid Q) \mid R &= ((P \parallel Q)/S_{PQ} \parallel R)/(A_{PQ} \cap A_R) && \text{(Prop. 24)}\\
&= ((P \parallel Q)/S_{PQ} \parallel R/S_{PQ})/(A_{PQ} \cap A_R) && \text{(Prop. 25(i) and } (*))\\
&= ((P \parallel Q) \parallel R)/S_{PQ}/(A_{PQ} \cap A_R) && \text{(Prop. 25(iii) and } (*))\\
&= ((P \parallel Q) \parallel R)/S_{PQR} && \text{(Prop. 25(ii) and } (**))\\
&= (P \parallel (Q \parallel R))/S_{PQR} && \text{(Thm. 12)}\\
&= P \mid (Q \mid R) && \text{(symmetrically)} \quad \square
\end{aligned}$$

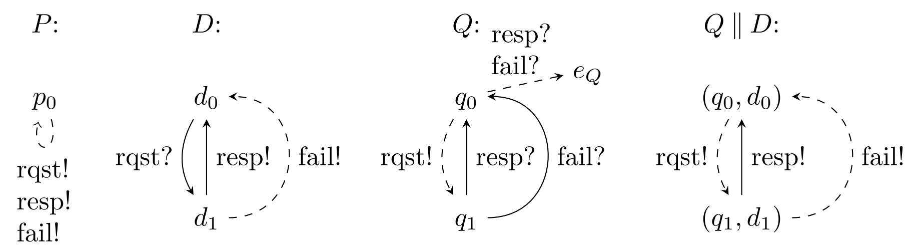

Figure 5: $Q = P /\!\!/ D$ with $q_0 = p_0 /\!\!/ d_0$ and $q_1 = p_0 /\!\!/ d_1$, where the alphabets are $A_P = \emptyset/\{\text{rqst, resp, fail}\}$, $A_D = \{\text{rqst}\}/\{\text{resp, fail}\}$, $A_Q = \{\text{resp, fail}\}/\{\text{rqst}\}$ and $A_{Q\|D} = \emptyset/\{\text{rqst, resp, fail}\}$.

4. Quotienting

The quotient operation is a kind of inverse or adjoined operation to parallel composition. It equips the theory with a means for component reuse and incremental, component-based specification. Given MIAs P and D, the quotient is the coarsest MIA Q such that $Q \parallel D \sqsubseteq P$ holds; we call this inequality the *defining inequality of the quotient*, and write $P /\!\!/ D$ if the quotient exists. In the following, we call P the *specification*, D the *divisor* (one might think of it as an already implemented component) and Q the *quotient* (the completion of D).

We demonstrate quotienting with the simple client-server application shown in Fig. 5, consisting of a given server D and one client. D can receive a request and answers with a response or possibly a failure message. The client should obviously have the outputs of D as inputs and D's inputs as outputs. So the parallel composition of server and client has only outputs, and it is error-free if and only if it is not universal. Thus, it must refine the specification P, also displayed in the figure. A most general, i.e., coarsest, specification for the client is then obtained as the quotient $Q =_{\text{df}} P /\!\!/ D$.

Fig. 5 gives a preview of this Q according to our construction below. Q may implement the sending of a request, and if so, it must be receptive for a response and a failure. If one of the latter two transitions were of may-modality, this would cause a communication mismatch in the parallel composition with D. The may-transitions resp? and fail? from q_0 to e_Q only exist to make Q as coarse as possible; they disappear in the parallel composition with D. Now, it is easy to check that the defining inequality $Q \| D \sqsubseteq P$ is satisfied. The example also shows that, in general, we do not have equality of $(P /\!\!/ D) \parallel D$ and P.

We define the quotient for a restricted set of MIAs, namely where the specification P has no τs and where the divisor D is may-deterministic and without τs. We call D *may-deterministic* if $d \overset{\alpha}{\dashrightarrow} d'$ and $d \overset{\alpha}{\dashrightarrow} d''$ implies $d' = d''$ for all d, d', d'' and α. Due to syntactic consistency, a may-deterministic MIA has no disjunctive must-transitions, i.e., the target sets of must-transitions are singletons. In addition, we exclude the pathological case where P has some state p and input i with $p \overset{i}{\dashrightarrow} e_P$ and $\exists p' \neq e_P.\, p \overset{i}{\dashrightarrow} p'$. Recall that transitions $p \overset{i}{\dashrightarrow} e_P$ are meant to express the following situation: (a) input i is not specified at p, but at the same time (b) p shall be refinable as in Interface Automata [4] by a state with an i-transition and arbitrary subsequent behaviour.

In the following, we call MIAs P and D satisfying our restrictions a *quotient pair*. Despite the restrictions, our quotient significantly generalises the one of Modal Interfaces [10], which considered deterministic specifications and deterministic divisors only.

4.1. Definition and Main Result

Like most other operators we define the quotient in two stages, where we write $\mathrm{may}_P(p, \alpha)$ for $\{p' \in P \mid p \overset{\alpha}{\dashrightarrow}_P p'\}$. Regarding the choice of the input and output alphabets in the following definition we adopt the one by Chilton et al. [7] and Raclet et al. [10]; we discuss alternative choices in Sec. 4.2.

Definition 27 (Pseudo-Quotient). *Let* $(P, I_P, O_P, \longrightarrow_P, \dashrightarrow_P, p_0, e_P)$ *and* $(D, I_D, O_D, \longrightarrow_D, \dashrightarrow_D, d_0, e_D)$ *be a quotient pair with* $A_D \subseteq A_P$ *and* $O_D \subseteq O_P$. *We set* $I =_{df} I_P \cup O_D$ *and* $O =_{df} O_P \setminus O_D$. *The* pseudo-quotient *of* P *over* D *is defined as the universal MIA* $(\{(e_P, e_D)\}, I, O, \emptyset, \emptyset, (e_P, e_D), (e_P, e_D))$ *if* $p_0 = e_P$. *Otherwise,* $P \oslash D =_{df} (P \times D, I, O, \longrightarrow, \dashrightarrow, (p_0, d_0), (e_P, e_D))$, *where the transition relations are defined by the following rules:*

$$\begin{array}{lll}
\text{(QMust1)} & (p,d) \xrightarrow{a} P' \times \{d\} & \textit{if } p \xrightarrow{a}_P P' \textit{ and } a \notin A_D \\
\text{(QMust2)} & (p,d) \xrightarrow{a} P' \times \{d'\} & \textit{if } p \xrightarrow{a}_P P' \textit{ and } d \xrightarrow{a}_D d' \\
\text{(QMust3)} & (p,d) \xrightarrow{a} P' \times \{d'\} & \textit{if } P' =_{df} \mathrm{may}_P(p,a) \neq \emptyset,\ e_P \notin P', \\
 & & \quad d \overset{a}{\dashrightarrow}_D d' \textit{ and } a \in O_D \\
\text{(QMay1)} & (p,d) \overset{a}{\dashrightarrow} (p', d) & \textit{if } p \overset{a}{\dashrightarrow}_P p' \neq e_P \textit{ and } a \notin A_D \\
\text{(QMay2)} & (p,d) \overset{a}{\dashrightarrow} (p', d') & \textit{if } p \overset{a}{\dashrightarrow}_P p' \neq e_P \textit{ and } d \xrightarrow{a}_D d' \\
\text{(QMay3)} & (p,d) \overset{a}{\dashrightarrow} (p', d') & \textit{if } p \overset{a}{\dashrightarrow}_P p',\ e_P \notin \mathrm{may}_P(p,a), \\
 & & \quad d \overset{a}{\dashrightarrow}_D d' \textit{ and } a \notin O_P \cap I_D \\
\text{(QMay4)} & (p,d) \overset{a}{\dashrightarrow} (e_P, e_D) & \textit{if } e_P \in \mathrm{may}_P(p,a) \quad \textit{(note: } a \in I_P \subseteq I\textit{)} \\
\text{(QMay5)} & (p,d) \overset{a}{\dashrightarrow} (e_P, e_D) & \textit{if } p \neq e_P,\ d \overset{a}{\not\dashrightarrow}_D \textit{ and } a \in A_D \setminus (O_P \cap I_D) \\
 & & \quad \textit{(note: } A_D \setminus (O_P \cap I_D) = I \cap A_D\textit{)}
\end{array}$$

The intuition behind a state (p, d) in $P \oslash D$ is that (p, d) composed in parallel with d refines state p, and that (p, d) should be the coarsest state wrt. MIA refinement satisfying this condition. With this in mind, we now justify the above rules intuitively. A formal proof is given in Lem. 29 and Thm. 30 below.

Rule (QMust1) is necessary due to the following consideration. If P has an a-must-transition where a is unknown to D, this can only originate from an a-must-transition in the quotient Q that we wish to construct; in order to be most permissive, each $p' \in P'$ must have a match in $Q \parallel D$. The corresponding consideration is true for Rule (QMay1), which also ensures syntactic consistency for Rule (QMust1).

Rule (QMust2) is obvious in the light of the choice of alphabet in Def. 27. As $P \oslash D$ has all actions of P and D in its alphabet, it also needs an a-must-transition to produce such a transition at $(p, d) \parallel d$. Here, Rule (QMay2) is the companion rule for guaranteeing syntactic consistency.

Rule (QMust3) ensures that (p,d) and d are compatible in case of an output of d. An application of this rule can be seen in Fig. 5 for action fail? at $q_1 = p_0 /\!\!/ d_1$. Syntactic consistency results from Rules (QMay2) and (QMay3); note that $a \in O_D$ implies $a \notin I_D$.

Observe how Rules (QMay2) and (QMay3) play together well. By the condition $a \notin O_P \cap I_D = O \cap I_D$, Rule (QMay3) does not generate an output a-may-transition in the pseudo-quotient that could make (p,d) and d illegal. These transitions are added by Rule (QMay2) if the a-transition at d is of must-modality and compatibility is ensured. This is exactly the situation in Fig. 5 for action rqst! at $q_0 = p_0 /\!\!/ d_0$.

Rule (QMay4) deals with the universal state in P. Obviously, $e_{P \oslash D} = (e_P, e_D)$ is the most general state of $P \oslash D$ that refines e_P in parallel composition with d. Implicitly, this rule replaces all states (e_P, d) by $e_{P \oslash D}$.

Rule (QMay5) makes $P \oslash D$ as coarse as possible. The input a-may-transitions introduced here just disappear in $(P \oslash D) \parallel D$, since a is blocked by D. This can be seen in Fig. 5 for actions resp? and fail? at $q_0 = p_0 /\!\!/ d_0$ and in $Q \parallel D$ at (q_0, d_0).

$P \oslash D$ is indeed a MIA. We have already argued for syntactic consistency. All rules ensure $p \neq e_P$; hence, $e_{P \oslash D}$ has no outgoing transitions. Incoming transitions of $e_{P \oslash D}$ can only arise from Rule (QMay4) or (QMay5), which are only applicable for $a \in I$.

Up to now we have only defined the pseudo-quotient. Considering a candidate pair (p,d), it may be impossible that p is refined by a state resulting from a parallel composition with d; this depends, e.g., on the modalities and the labels of the transitions leaving p and d. We call such pairs *impossible states* and remove them from the pseudo-quotient. For example, consider states $p \xrightarrow{a}$ and $d \overset{a}{\dashrightarrow}$ such that $d \overset{a}{\not\longrightarrow}$; no parallel composition with d refines p. While may-transitions can be refined by removing them and disjunctive transitions can be refined to subsets of their targets in order to prevent the reachability of impossible states, all states having a must-transition to only impossible states must also be removed. This pruning results in the quotient.

Definition 28 (Quotient)**.** *Let $P \oslash D$ be the pseudo-quotient of P over D. The set $G \subseteq P \times D$ of* impossible states *is defined as the least set satisfying the following rules:*

(G1)	$p \xrightarrow{a}_P$ *and* $d \overset{a}{\not\longrightarrow}_D$ *and* $a \in A_D$	*implies*	$(p,d) \in G$
(G2)	$p \neq e_P$ *and* $p \overset{a}{\not\dashrightarrow}_P$ *and* $d \overset{a}{\dashrightarrow}_D$ *and* $a \in O_D$	*implies*	$(p,d) \in G$
(G3)	$p \neq e_P$ *and* $d = e_D$	*implies*	$(p,d) \in G$
(G4)	$(p,d) \xrightarrow{a}_{P \oslash D} R'$ *and* $R' \subseteq G$	*implies*	$(p,d) \in G$

The quotient $P /\!\!/ D$ *is obtained by deleting all states $(p,q) \in G$ from $P \oslash D$. This also removes any may- or must-transition exiting a deleted state and any may-transition entering a deleted state; in addition, deleted states are removed from targets of disjunctive must-transitions. If $(p,d) \in P /\!\!/ D$, then we write $p /\!\!/ d$. If $(p_0, d_0) \notin P /\!\!/ D$, then the quotient P over D is not defined.*

Rule (G1) is obvious since (p, d) cannot ensure that $p \xrightarrow{a}_P$ is matched if d has no a-must-transition, as an a-may-transition or even a forbidden a at d can in no case compose to a refinement of a must-transition at p. Rule (G2) captures the situation where d has an output a that is forbidden at p. Offering an a-must-input in the quotient would lead to a transition in the parallel composition with d, while not offering a would lead to an error; both would not refine p. Rule (G3) captures the division by e_D: state e_D in parallel with any state is universal and does not refine $p \neq e_P$. Finally, Rule (G4) propagates back all impossibilities that cannot be avoided by refining.

Since $P \oslash D$ is a MIA, $P /\!\!/ D$ (i.e., the quotient is defined) is a MIA as well: syntactic consistency and the universal state are preserved by pruning. If the target set of a disjunctive must-transition became empty due to pruning, i.e., $R' \subseteq G$, Rule (G4) would be applicable and the source state and its must-transition are deleted. For the sink condition, observe the notes in parentheses in Rules (QMay4) and (QMay5).

We show next that the quotient operation above yields the coarsest MIA satisfying the defining inequality. For this proof, the next lemma ensures that the definedness of $\|$ and the definedness of $/\!\!/$ are mutually preserved across refinement.

Lemma 29. *Let P, D and Q be MIAs such that P and D is a quotient pair, $A_D \subseteq A_P$, $O_D \subseteq O_P$, $O_Q = O_P \setminus O_D$ and $I_Q = I_P \cup O_D$. Further, let p, d, q be states in P, D, Q, resp. Then, the following statements hold:*

1. *If $q \| d \sqsubseteq p$, then $p /\!\!/ d$ is defined.*
2. *If $q \sqsubseteq p /\!\!/ d$ and $p \neq e_P$, then $q \| d$ is defined.*

Proof. We write $\longrightarrow_\otimes$, $\longrightarrow_\|$, $\longrightarrow_\oslash$ and $\longrightarrow_{/\!\!/}$ as a shorthand for $\longrightarrow_{Q \otimes D}$, $\longrightarrow_{Q \| D}$, $\longrightarrow_{P \oslash D}$ and $\longrightarrow_{P /\!\!/ D}$, resp., and analogously for may-transitions. We show both claims by contraposition.

Claim 1: For all $(p, d) \in G$, the refinement $q \| d \sqsubseteq p$ does not hold for any $q \in Q$, possibly because $q \| d$ is not defined, i.e., $(q, d) \in E$ according to Def. 8. We prove this by induction on the derivation length according to the G-rules. In each case, we assume $q \| d \sqsubseteq p$ for some $q \in Q$ and derive a contradiction.

- **(G1)** $p \xrightarrow{a}$, $d \not\xrightarrow{a}$ and $a \in A_D$: By $q \| d \sqsubseteq p$, we have $q \| d \xrightarrow{a}_\|$, which can only be due to (PMust2) or (PMust3); thus, $d \xrightarrow{a}$, which is a contradiction.
- **(G2)** $p \neq e_P$, $p \overset{a}{\not\dashrightarrow}$, $d \overset{a}{\dashrightarrow}$ and $a \in O_D$: By $q \| d \sqsubseteq p$, we have $q \| d \overset{a}{\not\dashrightarrow}_\|$. Now, either $(q, d) \overset{a}{\dashrightarrow}_\otimes$ reaching an illegal state or $q \overset{a}{\not\dashrightarrow}$; in either case, $(q, d) \in E$, which is a contradiction.
- **(G3)** $p \neq e_P$ and $d = e_D$: Here, $(q, d) \in E$ is an inherited error, which is a contradiction.
- **(G4)** $(p, d) \xrightarrow{a}_\oslash R'$ with $R' \subseteq G$: Our claim holds for all $(p', d') \in R'$ by induction hypothesis, and the transition is due to one of the (QMust) rules:

(QMust1) $p \overset{a}{\longrightarrow} P'$, $a \notin A_D$ and $R' = P' \times \{d\}$: By $q \parallel d \sqsubseteq p$, we have $q \parallel d \overset{a}{\longrightarrow}_{\parallel} Q' \times \{d\}$ such that $\forall q'{\in}Q'\, \exists p'{\in}P'.\, q' \parallel d \sqsubseteq p'$. This is a contradiction, since $(p', d) \in R'$.

(QMust2) $p \overset{a}{\longrightarrow} P'$, $d \overset{a}{\longrightarrow} d'$ and $R' = P' \times \{d'\}$: $q \parallel d \sqsubseteq p$ implies the existence of a Q' with $q \overset{a}{\longrightarrow} Q'$ and $\forall q'{\in}Q'\, \exists p'{\in}P'.\, q' \parallel d' \sqsubseteq p'$. This is again a contradiction since $(p', d') \in R'$.

(QMust3) $e_P \notin \mathrm{may}_P(p, a) \neq \emptyset$, $R' = \mathrm{may}_P(p, a) \times \{d'\}$, $d \overset{a}{\dashrightarrow} d'$ and $a \in O_D$: Since $q \parallel d$ is defined, we have some $q \overset{a}{\longrightarrow} Q'$; otherwise, we would have $(q, d) \in E$. Thus, by the definition of illegal states, also $q' \parallel d'$ must be defined for some (and in fact all) $q' \in Q'$. Now, $q \parallel d \overset{a}{\dashrightarrow}_{\parallel} q' \parallel d'$ must be matched by some $p \overset{a}{\dashrightarrow} p'$ due to $q \parallel d \sqsubseteq p$, and we have $q' \parallel d' \sqsubseteq p'$. This is again a contradiction as $(p', d') \in R'$.

Claim 2: For all $(q, d) \in E$, $q \sqsubseteq p /\!\!/ d$ does not hold for any $p \in P$ with $p \neq e_P$, possibly because $p /\!\!/ d$ is not defined. We prove this by induction on the length of a local path from (q, d) to an error in $Q \otimes D$; here, all actions on the path are outputs. In each case, we assume $q \sqsubseteq p /\!\!/ d$ for some $p \in P$ with $p \neq e_P$ and derive a contradiction.

(Base) Let (q, d) be an error according to Def. 8.

(a) $q \overset{a}{\dashrightarrow} q'$, $d \overset{a}{\not\longrightarrow}$ and $a \in O_Q \cap I_D$: Here, $q \sqsubseteq p /\!\!/ d$ implies a transition $(p, d) \overset{a}{\dashrightarrow}_{\oslash} (p', d')$. But, such a transition cannot exist since none of the (QMay) rules applies; note that $a \in O_P \cap I_D$ for (QMay3) and (QMay5) and that $e_P \in \mathrm{may}_P(p, a)$ implies $a \in I_P$, which contradicts $a \in O_Q$, for (QMay4).

(b) $q \overset{a}{\not\longrightarrow}$, $d \overset{a}{\dashrightarrow} d'$ and $a \in I_Q \cap O_D$: As just noted, $a \in O_P$ implies $e_P \notin \mathrm{may}_P(p, a)$. Since (G2) does not apply, we have $\mathrm{may}_P(p, a) \neq \emptyset$. Thus, we get $p /\!\!/ d \overset{a}{\longrightarrow}_{/\!\!/}$ by (QMust3), contradicting $q \sqsubseteq p /\!\!/ d$ and $q \overset{a}{\not\longrightarrow}$.

(c) (q, d) is an inherited error: If $q = e_Q$, then $p /\!\!/ d = e_{P /\!\!/ D}$ by $q \sqsubseteq p /\!\!/ d$, and we have $p = e_P$. If $d = e_D$, then Rule (G3) and the definedness of $p /\!\!/ d$ imply $p = e_P$. Both cases contradict $p \neq e_P$.

(Step) Assume $(q, d) \overset{a}{\dashrightarrow}_{\otimes} (q', d') \in E$ with $a \in O_{Q \otimes D}$ such that our claim holds for (q', d') by induction. We consider the different rules that resulted in this transition.

(PMay1) $a \notin A_D$, $d' = d$ and $q \overset{a}{\dashrightarrow} q'$: By $q \sqsubseteq p /\!\!/ d$, there is a transition $p /\!\!/ d \overset{a}{\dashrightarrow}_{/\!\!/} p' /\!\!/ d''$ such that $q' \sqsubseteq p' /\!\!/ d''$. The only applicable Rule (QMay1) (note that $a \in O_P$) implies $d'' = d$ and $p' \neq e_P$. Thus, we have $q' \sqsubseteq p' /\!\!/ d$, contradicting the claim for (q', d').

(PMay2) $a \notin A_Q$, $q' = q$ and $d \overset{a}{\dashrightarrow} d'$: We have $a \in A_D \subseteq A_P = A_{P \oslash D} = A_Q$, which is a contradiction.

(PMay3) $q \overset{a}{\dashrightarrow} q'$ and $d \overset{a}{\dashrightarrow} d'$: By $q \sqsubseteq p /\!\!/ d$, there is a transition $p /\!\!/ d \overset{a}{\dashrightarrow}_{/\!\!/} p' /\!\!/ d''$ such that $q' \sqsubseteq p' /\!\!/ d''$. The only rules that are applicable are (QMay2) and (QMay3) (note that $a \in O_P$). Both rules imply $p' \neq e_P$ and, by may-determinism of D, $d'' = d'$. Thus, we have $q' \sqsubseteq p' /\!\!/ d'$, contradicting the claim for (q', d'). □

Theorem 30 ($/\!\!/$ is a Quotient Operator wrt. $\|$)**.** *Let P and D be a quotient pair and Q be a MIA such that $A_D \subseteq A_P$, $O_D \subseteq O_P$, $O_Q = O_P \setminus O_D$ and $I_Q = I_P \cup O_D$. Then, $Q \sqsubseteq P /\!\!/ D$ iff $Q \| D \sqsubseteq P$.*

Proof. We use the same shorthands as in Lem. 29. If $p_0 = e_P$, then $p_0 /\!\!/ d_0 = e_{P/\!\!/D}$ and both sides of the theorem's statement are simply true. For $p_0 \neq e_P$ we have: If $P /\!\!/ D$ is defined, then also $p_0 /\!\!/ d_0$ and, by Lem. 29, $q_0 \| d_0$ is defined. If $Q \| D \sqsubseteq P$, then the initial state of $Q \| D$ is $q_0 \| d_0 \neq e_{Q\|D}$ because of $p_0 \neq e_P$; with $q_0 \| d_0$ also $p_0 /\!\!/ d_0$ is defined by Lem. 29. Therefore, it suffices to establish the refinements.

"⇒": We show that $\mathcal{R} =_{\text{df}} \{(q\|d, p) \in (Q\|D) \times P \mid q \sqsubseteq p /\!\!/ d \text{ or } p = e_P\} \cup \{(e_{Q\|D}, e_P)\}$ is a MIA-refinement relation. We only have to consider a $(q\|d, p) \in \mathcal{R}$ with $p \neq e_P$. Note that Cases (iii) and (v) are mostly analogous to Cases (ii) and (iv), resp.

(i) From $p \neq e_P$ we conclude, by $q \sqsubseteq p /\!\!/ d$ and Lem. 29, that $q \| d$ exists, i.e., it is not the universal state.

(ii) $p \xrightarrow{i} P'$ for $i \in I_P$:

1. If $i \in A_D$ and $d \xrightarrow{i} d'$, then (QMust2) implies $(p, d) \xrightarrow{i}_{\oslash} P' \times \{d'\}$. In $P /\!\!/ D$, the target set might only be a subset $P'' \times \{d'\}$ of $P' \times \{d'\}$. By $q \sqsubseteq p /\!\!/ d$, we have $q \xrightarrow{i} Q'$ for some Q' such that $\forall q' \in Q' \, \exists p' \in P''.\, q' \sqsubseteq p' /\!\!/ d'$, whence $(q' \| d', p') \in \mathcal{R}$; note that $p' \neq e_P$ since, otherwise, $e_P \in P'$. Now, by (PMust3), there is a transition $(q, d) \xrightarrow{i}_{\otimes} Q' \times \{d'\}$. Since, for all $(q', d') \in Q' \times \{d'\}$, there is some $p' \in P''$ with $q' \sqsubseteq p' /\!\!/ d'$, we also have $q \| d \xrightarrow{i}_{\|} Q' \times \{d'\}$ by Lem. 29.

   To see the latter, note that it is impossible that $(q, d) \overset{i}{\dashrightarrow}_{\otimes} (\bar{q}, d') \in E$, for some $\bar{q} \in Q'$. This is because of the following reasons. If $(q, d) \overset{i}{\dashrightarrow}_{\otimes} (\bar{q}, d') \in E$, then $q \overset{i}{\dashrightarrow} \bar{q}$ by $I_P \subseteq I_Q$. Since $q \sqsubseteq p /\!\!/ d$, we have $p /\!\!/ d \overset{i}{\dashrightarrow}_{/\!\!/} \bar{p} /\!\!/ \bar{d}$ for some $\bar{p}$ with $\bar{q} \sqsubseteq \bar{p} /\!\!/ \bar{d}$, which can only be due to (QMay2). Observe that (QMay4) is excluded by P and D being a quotient pair, and that (QMay5) is excluded due to $d \overset{i}{\dashrightarrow}$. In the remaining case (QMay2) we have $p \overset{i}{\dashrightarrow} \bar{p} \neq e_P$ and $\bar{d} = d'$ due to may-determinism of D; further, Lem. 29 implies $(\bar{q}, d') \notin E$.

2. If $i \in A_D$ and $d \overset{i}{\not\longrightarrow}$, then $(p, d) \in G$ by (G1), which is impossible since $p /\!\!/ d$ is defined.

3. If $i \notin A_D$, the proof is analogous to Case 1 with $d = d'$, when replacing (QMust2) by (QMust1) and (PMust3) by (PMust1).

(iii) $p \overset{o}{\longrightarrow} P'$ for $o \in O_P$: Here, the same arguments as for (ii) apply.

(iv) $q \parallel d \overset{i}{\dashrightarrow}_{\parallel}$ and $i \in I_P = I_{Q\parallel D}$: Consider (a) $q \parallel d \overset{i}{\dashrightarrow}_{\parallel} q' \parallel d'$ or (b) $q \parallel d \overset{i}{\dashrightarrow}_{\parallel} e_{Q\parallel D}$ for $i \in I_{Q\parallel D}$. In both cases $(q, d) \overset{i}{\dashrightarrow}_{\otimes} (q', d')$ by one of (PMay1) or (PMay3), and $(q', d') \in E$ in case of (b). Rule (PMay2) is impossible as $A_Q = A_P \supseteq A_D$.

(PMay1) $q \overset{i}{\dashrightarrow} q'$ and $i \notin A_D$: We have $d = d'$, and $q \sqsubseteq p /\!\!/ d$ implies $p /\!\!/ d \overset{i}{\dashrightarrow}_{/\!\!/} p' /\!\!/ d''$ for some p', d'' such that $q' \sqsubseteq p' /\!\!/ d''$. Since $i \notin A_D$, we get either $d = d''$ and $p \overset{i}{\dashrightarrow} p' \neq e_P$ by (QMay1), or $p \overset{i}{\dashrightarrow} p' = e_P$ by (QMay4). In the latter case, we have $(q' \| d', e_P) \in \mathcal{R}$ for Case (a) and $(e_{Q\|D}, e_P) \in \mathcal{R}$ for Case (b). In the former case (QMay1), we have $(q' \| d', p') \in \mathcal{R}$ for Case (a) since $q' \sqsubseteq p' /\!\!/ d'$. Case (b) is impossible because $q' \parallel d' \notin E$ by Lem. 29, $q' \sqsubseteq p' /\!\!/ d'$ and $p' \neq e_P$.

(PMay3) $q \overset{i}{\dashrightarrow} q'$ and $d \overset{i}{\dashrightarrow} d'$: Since $q \sqsubseteq p /\!\!/ d$ we conclude $p /\!\!/ d \overset{i}{\dashrightarrow}_{/\!\!/} p' /\!\!/ d''$ for some p', d'' with $q' \sqsubseteq p' /\!\!/ d''$. This can be due to (QMay2), (QMay3) or (QMay4); in all cases we have $p \overset{i}{\dashrightarrow} p'$. In case (QMay4), we have $p' = e_P$ and $(q' \| d', e_P) \in \mathcal{R}$ for Case (a) and $(e_{Q\|D}, e_P) \in \mathcal{R}$ for Case (b). In the other Cases, we have $d'' = d'$ by may-determinism and $p' \neq e_P$; the proof now concludes like Case (QMay1) above.

(v) $q \| d \overset{o}{\dashrightarrow}_{\parallel}$ and $o \in O_P = O_{Q\|D}$: This case is already covered by (iv)(a), where the subcase due to (QMay4) does not apply.

"$\Leftarrow$": We show that $\mathcal{R} =_{\text{df}} \{(q, p /\!\!/ d) \in Q \times (P /\!\!/ D) \mid q \parallel d \sqsubseteq p \text{ or } p /\!\!/ d = e_{P /\!\!/ D}\}$ is a MIA-refinement relation. It suffices to consider some $(q, p /\!\!/ d) \in \mathcal{R}$ with $p /\!\!/ d \neq e_P /\!\!/ e_D$. In the following, the arguments for (iii) are analogous to those for (ii).

(i) Since $(q, d) \notin E$, we have $q \neq e_Q$.

(ii) $p /\!\!/ d \overset{i}{\longrightarrow}_{/\!\!/} R' \subseteq P' \times \{d'\}$ for $i \in I_{P /\!\!/ D}$, where $(p, d) \overset{i}{\longrightarrow}_{\oslash} P' \times \{d'\}$ is due to one of the (QMust) rules, and R' consists of the possible states of $P' \times \{d'\}$. In the following, we use $A_P = A_Q$ throughout.

(QMust1) $p \overset{i}{\longrightarrow} P'$, $d = d'$ and $i \notin A_D$: By $q \parallel d \sqsubseteq p$, we hav a transition $q \parallel d \overset{i}{\longrightarrow}_{\parallel} Q' \times \{d''\}$ for some Q', d'' with $\forall q' \in Q'\, \exists p' \in P'.\, q' \parallel d'' \sqsubseteq p'$. Since $i \notin A_D$, this transition can only be due to Rule (PMust1) and $d'' = d$. By Lem. 29, $q' \| d \sqsubseteq p'$ implies that $p' /\!\!/ d$ is not impossible, hence $p' /\!\!/ d \in R'$. Thus, we are done due to $q \overset{i}{\longrightarrow} Q'$.

(QMust2) $p \xrightarrow{i} P'$ and $d \xrightarrow{i} d'$: By $q \parallel d \sqsubseteq p$, we get $q \parallel d \xrightarrow{i}_{\parallel} Q' \times \{d'\}$ for some Q' such that $\forall q' {\in} Q'\, \exists p' {\in} P'.\, q' \parallel d' \sqsubseteq p'$. The transition must result from (PMust3). Thus, we are done as in (QMust1).

(QMust3) $P' = \mathrm{may}_P(p, a)$ and $d \overset{i}{\dashrightarrow} d'$ with $i \in O_D$: Because $q \parallel d$ is defined and $i \in I_Q \cap O_D$, we have $q \xrightarrow{i} Q'$ for some Q'. Now, Rule (PMay3) gives us $(q, d) \overset{i}{\dashrightarrow}_{\otimes} (q', d')$ for all $q' \in Q'$. Since $i \in O_{Q \otimes D}$ and $(q, d) \notin E$, we also know that $(q', d') \notin E$, hence $q \parallel d \overset{i}{\dashrightarrow}_{\parallel} q' \parallel d'$. By $q \parallel d \sqsubseteq p$ we have $\forall q' \in Q' \exists p' \in P'.\, p \overset{i}{\dashrightarrow} p'$ and $q' \parallel d' \sqsubseteq p'$. As above, $p' /\!\!/ d' \in R'$ and $q \xrightarrow{i} Q'$ matches $p /\!\!/ d \xrightarrow{i}_{/\!\!/} R'$.

(iii) $p /\!\!/ d \xrightarrow{o}_{/\!\!/} R'$ with $o \in O_{P /\!\!/ D} = O_P \setminus O_D$: The same arguments as for (ii) apply, except that Rule (QMust3) is not applicable due to $o \notin O_D$.

(iv) $q \overset{i}{\dashrightarrow} q'$ for $i \in I_Q$:

1. $i \notin A_D$: By (PMay1) we have $(q, d) \overset{i}{\dashrightarrow}_{\otimes} (q', d)$. Thus, either $q \| d \overset{i}{\dashrightarrow}_{\parallel} e_{Q \parallel D}$ or $q \| d \overset{i}{\dashrightarrow}_{\parallel} q' \| d$. In the first case we get $p \overset{i}{\dashrightarrow} e_P$, because of $q \parallel d \sqsubseteq p$, and $(p, d) \overset{i}{\dashrightarrow}_{\oslash} (e_P, e_D)$ by (QMay4). Since (e_P, e_D) can never be impossible, we have $p /\!\!/ d \overset{i}{\dashrightarrow}_{/\!\!/} e_P /\!\!/ e_D$ and are done. For the second case, $q \parallel d \overset{i}{\dashrightarrow}_{\parallel} q' \parallel d$, we get $p \overset{i}{\dashrightarrow} p'$ for some p' with $q' \parallel d \sqsubseteq p'$, because of $q \parallel d \sqsubseteq p$. If $p \overset{i}{\dashrightarrow} e_P$, we conclude as above. Otherwise, we get $(p, d) \overset{i}{\dashrightarrow}_{\oslash} (p', d)$ by (QMay1). Lem. 29 implies the definedness of $p' /\!\!/ d$, hence $p /\!\!/ d \overset{i}{\dashrightarrow}_{/\!\!/} p' /\!\!/ d$, and we are done.
2. $i \in A_D$ and $d \overset{i}{\not\dashrightarrow}$: By $p \neq e_P$ and $i \in A_D \setminus O_Q = A_D \setminus (O_P \cap I_D)$, we get $(p, d) \overset{i}{\dashrightarrow}_{\oslash} (e_P, e_D)$ by (QMay5). Since (e_P, e_D) can never be impossible, we have $p /\!\!/ d \overset{i}{\dashrightarrow}_{/\!\!/} e_P /\!\!/ e_D$ and are done.
3. $i \in A_D$ and $d \overset{i}{\dashrightarrow} d'$: By (PMay3), a transition $(q, d) \overset{i}{\dashrightarrow}_{\otimes} (q', d')$ exists. Thus, either $q \parallel d \overset{i}{\dashrightarrow}_{\parallel} e_{Q \parallel D}$ (only possible, if $i \in I_D$) or $q \parallel d \overset{i}{\dashrightarrow}_{\parallel} q' \parallel d'$ (ensured, if $i \in O_D$, since $q \parallel d$ defined). The first case is as in Case (iv).1. Also the second case is analogue to Case (iv).1, except for (QMay3) instead of (QMay1); for this, note that $i \in I_D$ implies $i \notin O_P$ by $i \in I_Q$.

(v) $q \overset{o}{\dashrightarrow} q'$ for $o \in O_Q$:

1. $o \in A_D$: We have $d \xrightarrow{o} d'$ for some d'; otherwise, $q \parallel d$ would not exist. By (PMay3) we have $(q, d) \overset{o}{\dashrightarrow}_{\otimes} (q', d')$, and hence

$q \parallel d \overset{o}{\dashrightarrow}_{\parallel} q' \parallel d'$ by definedness of $q \parallel d$. By $q \parallel d \sqsubseteq p$, we obtain $p \overset{o}{\dashrightarrow} p'$ for some p' with $q' \parallel d' \sqsubseteq p'$. Since $o \in O_P$, we conclude $p' \neq e_P$ and can then apply (QMay2) to get $(p, d) \overset{o}{\dashrightarrow}_{\oslash} (p', d')$. Lem. 29 implies the definedness of $p' /\!\!/ d'$, hence $p /\!\!/ d \overset{o}{\dashrightarrow}_{/\!\!/} p' /\!\!/ d'$ and we are done.

2. $o \notin A_D$: $q \parallel d \overset{o}{\dashrightarrow}_{\parallel} q' \parallel d$ by (PMay1) and definedness of $q \parallel d$; hence, due to $q \parallel d \sqsubseteq p$, there is a $p \overset{o}{\dashrightarrow} p'$ for some p' with $q' \parallel d \sqsubseteq p'$. Now we are done as in Case (v).1, applying (QMay1) instead of (QMay2). $\square$

From this theorem we can also conclude that $/\!\!/$ is monotonous wrt. $\sqsubseteq$ in the left argument.

Theorem 31 (Monotonicity of $/\!\!/$ wrt. $\sqsubseteq$)**.** *Let P_1, P_2, D be MIAs with $P_1 \sqsubseteq P_2$. If $P_1 /\!\!/ D$ is defined and P_2 and D are a quotient pair, then $P_2 /\!\!/ D$ is defined and $P_1 /\!\!/ D \sqsubseteq P_2 /\!\!/ D$.*

Proof. If $P_1 /\!\!/ D$ is defined, then $(P_1 /\!\!/ D) \parallel D \sqsubseteq P_1$ by Thm. 30. Applying the assumption $P_1 \sqsubseteq P_2$, transitivity of $\sqsubseteq$ and Thm. 30 again, we conclude that $P_1 /\!\!/ D \sqsubseteq P_2 /\!\!/ D$; in particular, $P_2 /\!\!/ D$ is also defined. $\square$

4.2. Discussion

We conclude this section by discussing the choice of alphabet for the quotient $Q = P /\!\!/ D$, argue why its input alphabet may be chosen differently, and conclude with some remarks on quotienting for Modal Interfaces (MI) [10] and Modal Transition Systems [20].

For $Q \parallel D \sqsubseteq P$ to hold, $Q \parallel D$ and P must have the same input alphabet and the same output alphabet. Thus, we have $O_Q = O_P \setminus O_D$ and $I_Q \supseteq I_P \setminus I_D$. Concerning the actions of D, quotient Q may listen to them but does not have to. Hence, $I_Q \subseteq (I_P \setminus I_D) \cup A_D = I_P \cup O_D$. The more inputs Q has, the easier it is to supply the behaviour ensuring $Q \parallel D \sqsubseteq P$. Thus, we have chosen the largest possible input alphabet $I_P \cup O_D$ for our quotient $P /\!\!/ D$, just as in [28] and [10]. When comparing some Q to $P /\!\!/ D$ in Thm. 30, Q necessarily has the same input and output alphabets as $P /\!\!/ D$, by Def. 4.

Quotient operators for interface theories are also discussed by Raclet [19], Raclet et al. [10] and Chilton et al. [7]. Our quotient $Q = P /\!\!/ D$ is most similar to the one in MI [10], where D is assumed to be may-deterministic, P and D have no internal transitions, and $I_Q = I_P \cup O_D$. However, P must also be may-deterministic in [10], whereas we additionally allow nondeterminism and disjunctive must-transitions in P.

In addition, we have corrected some technical shortcomings of MI. MI adapts the quotient operation for Modal Specifications from [19], with some additional rules defining the input and output alphabets of the quotient interface. However, compatibility is ignored for the quotient operation, which in [10] is an inverse or adjoint to their parallel product but *not* to parallel composition. This has been recognised in a technical report [16]. Unfortunately, that report employs a changed setting without the state tt as in [10] or a universal state as in our work. This is reflected by a different, non-compositional parallel composition that does not allow arbitrary behaviour in case of an inconsistency and that employs a more aggressive pruning strategy, where a mismatch can also occur if two systems share an input.

Beneš et al. [20] investigate quotienting for nondeterministic specifications in the settings of Modal Transition Systems (MTS) and Nondeterministic Acceptance Automata (NAA). They address nondeterminism by constructing sets of possible next quotient states for each transition label. In principle, a similar solution would be necessary in order to relax our determinism requirement on the denominator. However, it is not straightforward to adopt this solution in the context of internal transitions and input/output with the related compatibility issues, which are core ingredients of interface theories being present since the very first publications on IA by de Alfaro and Henzinger. Not considering input/output simplifies the quotient because a significantly simpler composition operator is involved, which corresponds more to our parallel product than our parallel composition. In addition, Beneš et al. assume a single global alphabet and do not consider alphabet extension, which is particularly difficult for the quotient, as discussed in Sec. 6.

5. Conjunction and Disjunction

Besides parallel composition and quotienting, conjunction is one of the most important operators of interface theories. It allows one to specify different perspectives of a system separately, from which an overall specification can be determined by conjunctive composition. More formally, the conjunction should be the coarsest specification that refines the given perspective specifications, i.e., it should characterise the greatest lower bound of the refinement preorder. In the following, we define conjunction for MIAs with common alphabets, as we did for MIA refinement. Similar to parallel composition, we first present a conjunctive product and, in a second step, remove state pairs with contradictory specifications.

Definition 32 (Conjunctive Product)**.** *Consider MIAs $(P, I, O, \longrightarrow_P, \dashrightarrow_P, p_0, e_P)$ and $(Q, I, O, \longrightarrow_Q, \dashrightarrow_Q, q_0, e_Q)$ with common alphabets. The* conjunctive product *is defined as $P \& Q =_{df} (P \times Q, I, O, \longrightarrow, \dashrightarrow, (p_0, q_0), (e_P, e_Q))$ by the following operational transition rules:*

$$
\begin{array}{lll}
\text{(OMust1)} & (p,q) \xrightarrow{\omega} \{(p',q') \,|\, p' \in P',\ q \stackrel{\hat{\omega}}{\Longrightarrow}_Q q'\} & \textit{if } p \xrightarrow{\omega}_P P' \textit{ and } q \stackrel{\hat{\omega}}{\Longrightarrow}_Q \\
\text{(OMust2)} & (p,q) \xrightarrow{\omega} \{(p',q') \,|\, p \stackrel{\hat{\omega}}{\Longrightarrow}_P p',\ q' \in Q'\} & \textit{if } p \stackrel{\hat{\omega}}{\Longrightarrow}_P \textit{ and } q \xrightarrow{\omega}_Q Q' \\
\text{(IMust1)} & (p,q) \xrightarrow{i} \{(p',q') \,|\, p' \in P',\ q \stackrel{i}{\dashrightarrow}\stackrel{\varepsilon}{\Longrightarrow}_Q q'\} & \textit{if } p \xrightarrow{i}_P P' \textit{ and } q \stackrel{i}{\dashrightarrow}_Q \\
\text{(IMust2)} & (p,q) \xrightarrow{i} \{(p',q') \,|\, p \stackrel{i}{\dashrightarrow}\stackrel{\varepsilon}{\Longrightarrow}_P p',\ q' \in Q'\} & \textit{if } p \stackrel{i}{\dashrightarrow}_P \textit{ and } q \xrightarrow{i}_Q Q' \\
\text{(EMust1)} & (p, e_Q) \xrightarrow{\alpha} P' \times \{e_Q\} & \textit{if } p \xrightarrow{\alpha}_P P' \\
\text{(EMust2)} & (e_P, q) \xrightarrow{\alpha} \{e_P\} \times Q' & \textit{if } q \xrightarrow{\alpha}_Q Q'
\end{array}
$$

$$
\begin{tikzcd}
P\colon\; p_0 \arrow[r, "\tau"] \arrow[d, "o"] & p_1 \arrow[r, "\tau"] \arrow[d, "o"] & p_2 \arrow[d, "o"] & \cdots & & Q\colon\; q_0 \arrow[r, "o"] & q_1 \\
p_0' & p_1' & p_2' & \cdots & & &
\end{tikzcd}
$$

Figure 6: Example of a conjunction leading to a transition with an infinite target set.

(May1)	$(p, q) \overset{\tau}{\dashrightarrow} (p', q)$	*if* $p \overset{\tau}{\Longrightarrow}_P p'$
(May2)	$(p, q) \overset{\tau}{\dashrightarrow} (p, q')$	*if* $q \overset{\tau}{\Longrightarrow}_Q q'$
(OMay)	$(p, q) \overset{\omega}{\dashrightarrow} (p', q')$	*if* $p \overset{\omega}{\Longrightarrow}_P p'$ *and* $q \overset{\omega}{\Longrightarrow}_Q q'$
(IMay)	$(p, q) \overset{i}{\dashrightarrow} (p', q')$	*if* $p \overset{i}{\dashrightarrow}\overset{\varepsilon}{\Longrightarrow}_P p'$ *and* $q \overset{i}{\dashrightarrow}\overset{\varepsilon}{\Longrightarrow}_Q q'$
(EMay1)	$(p, e_Q) \overset{\alpha}{\dashrightarrow} (p', e_Q)$	*if* $p \overset{\alpha}{\dashrightarrow}_P p'$
(EMay2)	$(e_P, q) \overset{\alpha}{\dashrightarrow} (e_P, q')$	*if* $q \overset{\alpha}{\dashrightarrow}_Q q'$

Note that this definition is similar to the one in [12], except for the treatment of inputs and the universal state. The conjunctive product is inherently different from the parallel product: single transitions are defined through weak transitions, e.g., as in Rules (OMust), (IMust), (May), and τ-transitions synchronise by Rule (OMay). Furthermore, as given by Rules (EMust) and (EMay), the universal states are neutral elements for the conjunctive product, whereas they are absorbing for the parallel product.

Definition 33 (Conjunction)**.** *Given a conjunctive product $P \& Q$, the set $F \subseteq P \times Q$ of (logically)* inconsistent states *is defined as the least set satisfying the following rules for all $p \neq e_P$ and $q \neq e_Q$:*

(F1)	$p \overset{o}{\longrightarrow}_P$ *and* $q \overset{o}{\not\Longrightarrow}_Q$	*implies*	$(p, q) \in F$
(F2)	$p \overset{o}{\not\Longrightarrow}_P$ *and* $q \overset{o}{\longrightarrow}_Q$	*implies*	$(p, q) \in F$
(F3)	$p \overset{i}{\longrightarrow}_P$ *and* $q \overset{i}{\not\dashrightarrow}_Q$	*implies*	$(p, q) \in F$
(F4)	$p \overset{i}{\not\dashrightarrow}_P$ *and* $q \overset{i}{\longrightarrow}_Q$	*implies*	$(p, q) \in F$
(F5)	$(p, q) \overset{\alpha}{\longrightarrow} R'$ *and* $R' \subseteq F$	*implies*	$(p, q) \in F$

The conjunction $P \wedge Q$ is obtained (analogously to Def. 28) by deleting all states $(p, q) \in F$ from $P \& Q$. This also removes any may- or must-transition exiting a deleted state and any may-transition entering a deleted state; in addition, deleted states are removed from targets of disjunctive must-transitions. We write $p \wedge q$ for state (p, q) of $P \wedge Q$; all such states are defined – and consistent – by construction. However, if $(p_0, q_0) \in F$, then the conjunction of P and Q does not exist.

Note that the weak transitions in Rules (OMust) and (IMust) may lead to disjunctive transitions with infinite target sets, which were prohibited in [1]. For example, consider the conjunction of the MIAs P and Q depicted in Fig. 6. Infinitely many weak o-transitions start from p_0, yielding an infinite disjunctive

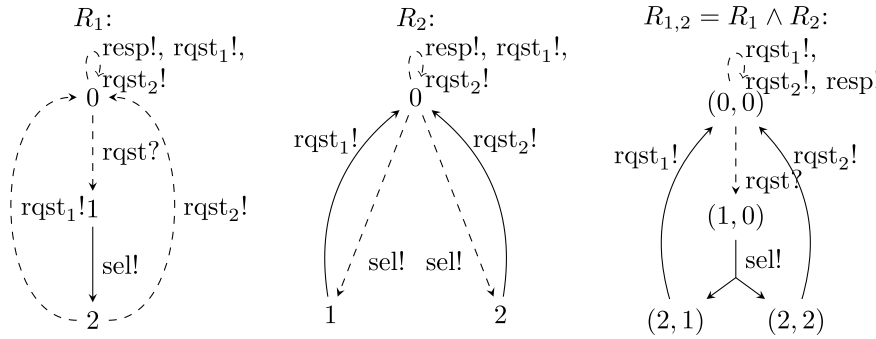

Figure 7: Conjunction of MIAs may lead to disjunctive transitions.

o-transition at $p_0 \wedge q_0$ due to Rule (OMust2). We addressed this issue by generalising MIAs (cf. Def. 1) to allow infinite target sets of must-transitions and by adapting Def. 2 accordingly. Note that the abovementioned problem arises only for an infinite-state MIA (cf. P in Fig. 6) and is not a problem in practice, where MIAs are expected to be finite state.

An example of conjunction is given in Fig. 7. It shows the requirement for a server front-end that shall route between a client and at least one of two possible back-ends. In practice, one might specify this requirement directly as MIA $R_{1,2}$. Here, we specify it as two separate MIAs R_1 and R_2 solely to illustrate conjunction. Requirement R_1 states that the selection (sel!) of a back-end must be made after a client's request is received (rqst?). After that selection, the only possibility is to redirect the request to one of the back-ends (rqst$_1$!, rqst$_2$!). The loops in state 0 are necessary in order to not constrain the corresponding actions overly, since they might be used by other requirements. Action resp! is included in R_1 to prevent an early abortion in states 1 and 2; otherwise, a response could be sent to the client before a back-end is contacted. Requirement R_2 makes sel! a true selection. Once the selection is made, the request is forwarded either to B_1 or to B_2 (rqst$_1$!, rqst$_2$!). Together, requirements R_1 and R_2 ensure that at least one of the back-ends will be contacted. This is expressed in the conjunction $R_{1,2} = R_1 \wedge R_2$, where the selection process (sel!) is given by a *disjunctive* must-transition, although none of the conjuncts has a disjunctive transition. This is due to the combination of modalities with nondeterminism and cannot be expressed in a deterministic theory, such as in Modal Interfaces [10] which our theory extends. Although one might approximate the disjunctive sel! by individual selection actions sel$_1$! and sel$_2$! for each back-end, the conjunction would either have both actions as may-transitions and, thus, allow one to omit both, or would have both actions as must-transitions, disallowing a server application with only one of the back-ends.

Next, we prove that conjunction as defined above is the greatest lower bound wrt. MIA refinement. To this end, we introduce the notion of a witness as in [12]:

Definition 34 (Witness)**.** *A witness W of $P \& Q$ is a subset of $P \times Q$ such that* *the following conditions hold for all* $(p,q) \in W$*:*

(W1)	$p \stackrel{o}{\longrightarrow}_P$	*implies*	$q \stackrel{o}{=\!\!\Rightarrow}_Q$ *or* $q = e_Q$
(W2)	$q \stackrel{o}{\longrightarrow}_Q$	*implies*	$p \stackrel{o}{=\!\!\Rightarrow}_P$ *or* $p = e_P$
(W3)	$p \stackrel{i}{\longrightarrow}_P$	*implies*	$q \stackrel{i}{\dashrightarrow}_Q$ *or* $q = e_Q$
(W4)	$q \stackrel{i}{\longrightarrow}_Q$	*implies*	$p \stackrel{i}{\dashrightarrow}_P$ *or* $p = e_P$
(W5)	$(p,q) \stackrel{\alpha}{\longrightarrow} R'$	*implies*	$R' \cap W \neq \emptyset$

Intuitively, a witness is a set of state pairs that are consistent and, thus, it witnesses the existence of a conjunction.

Lemma 35 (Concrete Witness)**.** *Let* P*,* Q *and* R *be MIAs with common alphabets.*

1. *For any witness* W *of* $P\&Q$*, we have* $F \cap W = \emptyset$*.*
2. *The set* $\{(p,q) \in P \times Q \mid \exists r \in R.\, r \sqsubseteq p$ *and* $r \sqsubseteq q\}$ *is a witness of* $P\&Q$*.*

Proof. While the first statement of the lemma is quite obvious, we prove here that $W =_{\mathrm{df}} \{(p,q) \in P \times Q \mid \exists r \in R.\, r \sqsubseteq p \text{ and } r \sqsubseteq q\}$ is a witness of $P\&Q$:

(W1) $p \stackrel{o}{\longrightarrow}_P P'$ implies $r \stackrel{o}{\Longrightarrow}_R R'$ for some R' by $r \sqsubseteq p$. Choose some $r' \in R'$. Then, $r \stackrel{o}{=\!\!\Rightarrow}_R r'$ by syntactic consistency, and $q \stackrel{o}{=\!\!\Rightarrow}_Q$ or $q = e_Q$ by $r \sqsubseteq q$.

(W2) Analogous to (W1).

(W3) Similar to (W1) with o replaced by i, $\Longrightarrow$ by $\longrightarrow\!\stackrel{\varepsilon}{\Longrightarrow}$, and $=\!\!\Rightarrow$ by $\dashrightarrow\!\stackrel{\varepsilon}{=\!\!\Rightarrow}$.

(W4) Analogous to (W3).

(W5) Consider $(p,q) \in W$ due to r, with $(p,q) \stackrel{\omega}{\longrightarrow} S'$ because of $p \stackrel{\omega}{\longrightarrow}_P P'$ and $S' = \{(p',q') \mid p' \in P',\, q \stackrel{\hat{\omega}}{=\!\!\Rightarrow}_Q q'\}$ by (OMust1). By $r \sqsubseteq p$ and $p \neq e_P$, we get an $R' \subseteq R$ with $r \stackrel{\hat{\omega}}{\Longrightarrow}_R R'$ and $\forall r' \in R'\, \exists p' \in P'.\, r' \sqsubseteq p'$. Choose $r' \in R'$; now, $r \stackrel{\hat{\omega}}{=\!\!\Rightarrow}_R r'$ due to syntactic consistency, and $q \stackrel{\hat{\omega}}{=\!\!\Rightarrow}_Q q'$ with $r' \sqsubseteq q'$ by $r \sqsubseteq q$; this also holds if $q = e_Q$ and $\omega = \tau$. Thus, we have $p' \in P'$ and q' such that $(p',q') \in S' \cap W$ due to r'. The same line of argument works for inputs with trailing-weak instead of weak transitions and using (IMust1) instead of (OMust1). The remaining case concerns transitions $(p, e_Q) \stackrel{\alpha}{\longrightarrow} S'$ because of $p \stackrel{\alpha}{\longrightarrow}_P P'$ and $S' = P' \times \{e_Q\}$ by (EMust1). Choose some $p' \in P'$; then, $(p', e_Q) \in S' \cap W$ due to $r \sqsubseteq p$. □

On the basis of this lemma we can now establish the desired greatest lower bound result for $\wedge$, which implies the compositionality of $\sqsubseteq$ wrt. $\wedge$ (cf. [12]).

Theorem 36 ($\wedge$ is And)**.** *Let* P *and* Q *be MIAs with common alphabets.*

1. $(\exists R.\ R \sqsubseteq P$ *and* $R \sqsubseteq Q)$ *iff* $P \wedge Q$ *is defined.*

2. If $P \wedge Q$ is defined, then $R \sqsubseteq P$ and $R \sqsubseteq Q$ iff $R \sqsubseteq P \wedge Q$, for any R.

Note that MIA R is implicitly required to have the same alphabets as P and Q, by Def. 4.

Proof. Claim 1 "$\Rightarrow$": This follows from Lem. 35.

Claims 1 and 2 "$\Leftarrow$": It suffices to show that $\mathcal{R} =_{\text{df}} \{(r,p) \mid \exists q.\, r \sqsubseteq p \wedge q\}$ is a MIA-refinement relation. Then, in particular, Claim 1 "$\Leftarrow$" follows by choosing $R = P \wedge Q$. Furthermore, note that (EMust1) and (EMay1) essentially produce an isomorphic copy of P. The refinement conditions for states $(r,p) \in \mathcal{R}$ due to $q = e_Q$ hold by definition of $\mathcal{R}$, and we can ignore these rules in the rest of this proof.

We check the conditions of Def. 4 for some $(r,p) \in \mathcal{R}$ due to q, where $p \neq e_P$:

- $p \neq e_P$ implies $p \wedge q \neq e_P \wedge e_Q$. By $r \sqsubseteq p \wedge q$, we have $r \neq e_R$.

- Let $p \xrightarrow{\alpha}_P P'$; then, $q \overset{\hat{\alpha}}{\Longrightarrow}_Q$. For $\alpha \neq \tau$, this is because, otherwise, $p \wedge q$ would not be defined due to (F1). Hence, by (OMust1) (or similarly (IMust1)), $p \wedge q \xrightarrow{\alpha} \{p' \wedge q' \mid p' \in P',\ q \overset{\hat{\alpha}}{\Longrightarrow}_Q q',\ p' \wedge q' \text{ defined}\}$. By $r \sqsubseteq p \wedge q$, we get $r \overset{\hat{\alpha}}{\Longrightarrow}_R R'$ such that $\forall r' {\in} R'\ \exists p' {\wedge} q'.\, p' \in P',\ q \overset{\hat{\alpha}}{\Longrightarrow}_Q q'$ and $r' \sqsubseteq p' {\wedge} q'$. Thus, $\forall r' {\in} R'\ \exists p' {\in} P'.\, (r',p') \in \mathcal{R}$.

- $r \overset{\alpha}{\dashrightarrow}_R r'$ implies $\exists p' {\wedge} q'.\, p {\wedge} q \overset{\hat{\alpha}}{\Longrightarrow} p' {\wedge} q'$ and $r' \sqsubseteq p' {\wedge} q'$. The contribution of p in this weak transition sequence gives $p \overset{\hat{\alpha}}{\Longrightarrow}_P p'$, and, thus, we have $(r',p') \in \mathcal{R}$ due to q'.

Claim 2 "$\Rightarrow$": Here, we show that $\mathcal{R} =_{\text{df}} \{(r, p \wedge q) \mid r \sqsubseteq p \text{ and } r \sqsubseteq q\}$ is a MIA-refinement relation; by Claim 1, $p \wedge q$ is defined whenever $r \sqsubseteq p$ and $r \sqsubseteq q$. As above, the (EMust) and (EMay) rules do not need to be checked, in particular, since $r' \sqsubseteq e_Q$ for all r'. We now verify the conditions of Def. 4:

- If $p \wedge q \neq e_P \wedge e_Q$, then w.l.o.g. $p \neq e_P$. By $r \sqsubseteq p$, we also have $r \neq e_R$.

- Let $p \wedge q \xrightarrow{\alpha} S'$; in case of $\alpha \in O \cup \{\tau\}$ and w.l.o.g., this is due to $p \xrightarrow{\alpha}_P P'$ and $S' = \{p' \wedge q' \mid p' \in P',\ q \overset{\hat{\alpha}}{\Longrightarrow}_Q q',\ p' \wedge q' \text{ defined}\}$. Because of $r \sqsubseteq p$, we have $r \overset{\hat{\alpha}}{\Longrightarrow}_R R'$ so that $\forall r' {\in} R'\ \exists p' {\in} P'.\ r' \sqsubseteq p'$. Consider some arbitrary $r' \in R'$ and the resp. $p' \in P'$. Then, $r \overset{\hat{\alpha}}{\Longrightarrow}_R r'$ by syntactic consistency and, due to $r \sqsubseteq q$ and Prop. 5, there exists some q' with $q \overset{\hat{\alpha}}{\Longrightarrow}_Q q'$ and $r' \sqsubseteq q'$. Thus, $p' \wedge q' \in S'$ and $(r', p' \wedge q') \in \mathcal{R}$. In case of $\alpha \in I$, we follow the same line of arguments, where we simply replace weak transitions by trailing-weak transitions.

- Let $r \overset{\alpha}{\dashrightarrow}_R r'$, for $\alpha \in O \cup \{\tau\}$, and consider $p \overset{\hat{\alpha}}{\Longrightarrow}_P p'$ and $q \overset{\hat{\alpha}}{\Longrightarrow}_Q q'$ satisfying $r' \sqsubseteq p'$ and $r' \sqsubseteq q'$. Thus, $(r', p' \wedge q') \in \mathcal{R}$. Further, if $\alpha \neq \tau$, we have $p \wedge q \overset{\alpha}{\dashrightarrow} p' \wedge q'$ by (OMay). Otherwise, either $p \overset{\tau}{\Longrightarrow}_P p'$ and

$q \overset{\tau}{\Longrightarrow}_Q q'$ and we are done by (OMay), or w.l.o.g. $p \overset{\tau}{\Longrightarrow}_P p'$ and $q = q'$ and we are done by (May1), or $p = p'$ and $q = q'$. In case of $\alpha \in I$, we follow the same line of arguments as for $\alpha \in O$, where we replace weak transitions by trailing-weak transitions and use (IMay) instead of (OMay). $\square$

Corollary 37. *MIA refinement is compositional wrt. conjunction.*

Clearly, conjunction is commutative. Furthermore, any conjunction operator that satisfies the statement of Thm. 36 for some preorder $\sqsubseteq$ is associative.

Lemma 38. *Let P, Q, R and S be MIAs.*

1. *$P \wedge (Q \wedge R)$ is defined iff $(P \wedge Q) \wedge R$ is defined.*
2. *If $P \wedge (Q \wedge R)$ is defined, then $S \sqsubseteq P \wedge (Q \wedge R)$ iff $S \sqsubseteq (P \wedge Q) \wedge R$.*

Proof. 1. Thm. 36.1, 36.2 imply that $P \wedge (Q \wedge R)$ is defined iff $\exists S.\, S \sqsubseteq P$ and $S \sqsubseteq Q \wedge R$ iff $\exists S.\, S \sqsubseteq P$ and $S \sqsubseteq Q$ and $S \sqsubseteq R$ iff $\exists S.\, S \sqsubseteq P \wedge Q$ and $S \sqsubseteq R$ iff $(P \wedge Q) \wedge R$ is defined. Claim 2 follows directly from multiple applications of Thm. 36.2. $\square$

As a consequence of Lem. 38 we obtain strong associativity of conjunction.

Theorem 39 (Associativity of Conjunction)**.** *Conjunction is associative in the sense that, if one of $P \wedge (Q \wedge R)$ and $(P \wedge Q) \wedge R$ is defined, then both are defined and $P \wedge (Q \wedge R) \sqsupseteq\sqsubseteq (P \wedge Q) \wedge R$.*

We now turn our attention to disjunction $\vee$ on MIAs with the same alphabets and show that $\vee$ corresponds to the least upper bound of MIA refinement. Disjunction may be used in systems design for expressing alternatives, e.g., in the context of product families.

Definition 40 (Disjunction)**.** *Given two MIAs $(P, I, O, \longrightarrow_P, \dashrightarrow_P, p_0, e_P)$ and $(Q, I, O, \longrightarrow_Q, \dashrightarrow_Q, q_0, e_Q)$ with common input and output alphabets. Writing also e for $e_P \vee e_Q$, the* disjunction *$P \vee Q$ is defined as $(\{e\}, I, O, \emptyset, \emptyset, e, e)$ if $p_0 = e_P$ or $q_0 = e_Q$. Otherwise, and assuming disjoint state sets, $P \vee Q = (\{p_0 \vee q_0, e\} \cup P \cup Q, I, O, \longrightarrow, \dashrightarrow, p_0 \vee q_0, e)$, where $\longrightarrow$ and $\dashrightarrow$ are the least sets satisfying the conditions $\longrightarrow_P \subseteq \longrightarrow$, $\dashrightarrow_P \subseteq \dashrightarrow$, $\longrightarrow_Q \subseteq \longrightarrow$, $\dashrightarrow_Q \subseteq \dashrightarrow$, and the following rules:*

(Must)	$p_0 \vee q_0 \xrightarrow{\tau} \{p_0, q_0\}$	*if* $p_0 \neq e_P$ *and* $q_0 \neq e_Q$
(IMust)	$p_0 \vee q_0 \xrightarrow{i} P' \cup Q'$	*if* $p_0 \xrightarrow{i}_P P'$ *and* $q_0 \xrightarrow{i}_Q Q'$
(May)	$p_0 \vee q_0 \overset{\tau}{\dashrightarrow} p_0,\ p_0 \vee q_0 \overset{\tau}{\dashrightarrow} q_0$	*if* $p_0 \neq e_P$ *and* $q_0 \neq e_Q$
(IMay1)	$p_0 \vee q_0 \overset{i}{\dashrightarrow} p'$	*if* $p_0 \overset{i}{\dashrightarrow}_P p'$
(IMay2)	$p_0 \vee q_0 \overset{i}{\dashrightarrow} q'$	*if* $q_0 \overset{i}{\dashrightarrow}_Q q'$

Further, for each input may-transition to e_P or e_Q, the target is replaced by e.

It is not difficult to see that disjunction is commutative and associative. The latter follows from the dual statement to Thm. 36, namely that $\vee$ is indeed disjunction.

Theorem 41 ($\vee$ is Or)**.** *Let P, Q and R be MIAs with common alphabets. Then, $P \vee Q \sqsubseteq R$ iff $P \sqsubseteq R$ and $Q \sqsubseteq R$.*

Proof. If, say, $p_0 = e_P$, then both sides imply $r_0 = e_R$, which implies $Q \sqsubseteq R$ in any case. So we can assume that neither $p_0 = e_P$ nor $q_0 = e_Q$.

"$\Longrightarrow$": We establish w.l.o.g. that $\mathcal{R} =_{\text{df}} \{(p_0, r) \mid p_0 \vee q_0 \sqsubseteq r\} \cup \sqsubseteq$ is a MIA-refinement relation. To do so, we let $(p_0, r) \in \mathcal{R}$ and check the conditions of Def. 4:

(i) If $r \neq e_R$, then $p_0 \vee q_0 \neq e$; thus, $p_0 \neq e_P$.

(ii) Let $r \xrightarrow{i}_R R'$. Because of $p_0 \vee q_0 \sqsubseteq r$ and by the only applicable Rule (IMust), we have $p_0 \vee q_0 \xrightarrow{i}\overset{\varepsilon}{\Longrightarrow} P' \cup Q'$, due to $p_0 \xrightarrow{i}\overset{\varepsilon}{\Longrightarrow}_P P'$ and $q_0 \xrightarrow{i}\overset{\varepsilon}{\Longrightarrow}_Q Q'$, such that $\forall p' \in P' \cup Q'\, \exists r' \in R'.\, p' \sqsubseteq r'$; recall $P \cap Q = \emptyset$. Hence, $\forall p' \in P'\, \exists r' \in R'.\, p' \sqsubseteq r'$ and, thus, $(p', r') \in \mathcal{R}$.

(iii) Let $r \xrightarrow{\omega}_R R'$. By $p_0 \vee q_0 \sqsubseteq r$, we get $p_0 \vee q_0 \overset{\hat{\omega}}{\Longrightarrow} S'$ for some S' such that $\forall s{\in}S'\ \exists r'{\in}R'.\ s \sqsubseteq r'$. If $p_0 \vee q_0 \overset{\omega}{\Longrightarrow} S'$, then the transition sequence underlying this weak transition starts with $p_0 \vee q_0 \xrightarrow{\tau} \{p_0, q_0\}$, and the remainder can be decomposed showing $p_0 \overset{\hat{\omega}}{\Longrightarrow}_P P'$, $q_0 \overset{\hat{\omega}}{\Longrightarrow}_Q Q'$ and $S' = P' \cup Q'$. Because $\forall p'{\in}P'\, \exists r'{\in}R'.\, p' \sqsubseteq r'$, we are done now. The only remaining case is $\omega = \tau$ and $S' = \{p_0 \vee q_0\}$, in which there is some $r' \in R'$ such that $p_0 \vee q_0 \sqsubseteq r'$, i.e., $(p_0, r') \in \mathcal{R}$. Hence, we are done in this case, too, since $p_0 \overset{\hat{\tau}}{\Longrightarrow}_P p_0$.

(iv) Let $p_0 \overset{i}{\dashrightarrow}_P p'$. Then, $p_0 \vee q_0 \overset{i}{\dashrightarrow} p'$ and, due to $p_0 \vee q_0 \sqsubseteq r$, we obtain some r' with $r \overset{i}{\dashrightarrow}\overset{\varepsilon}{\Longrightarrow}_R r'$ and $p' \sqsubseteq r'$ by Def. 4(iv).

(v) Let $p_0 \overset{\omega}{\dashrightarrow}_P p'$. Then, $p_0 \vee q_0 \overset{\tau}{\dashrightarrow} p_0$ and, due to $p_0 \vee q_0 \sqsubseteq r$, we apply Def. 4(v) twice to obtain some r' with $r \overset{\hat{\omega}}{\Longrightarrow}_R r'$ and $p' \sqsubseteq r'$.

"$\Longleftarrow$": We prove that $\mathcal{R} =_{\text{df}} \{(p_0 \vee q_0, r) \mid p_0 \sqsubseteq r \text{ and } q_0 \sqsubseteq r\} \cup \sqsubseteq$ is a MIA-refinement relation; consider $(p_0 \vee q_0, r)$ with $r \neq e_R$.

(i) Since $r \neq e_R$, we have $p_0 \neq e_P$ and $q_0 \neq e_Q$; thus, $p_0 \vee q_0 \neq e$.

(ii) Let $r \xrightarrow{i}_R R'$. By $p_0 \sqsubseteq r$ and $q_0 \sqsubseteq r$, we have P' and Q' satisfying $p_0 \xrightarrow{i}\overset{\varepsilon}{\Longrightarrow}_P P'$, $q_0 \xrightarrow{i}\overset{\varepsilon}{\Longrightarrow}_Q Q'$ such that $\forall p'{\in}P'\, \exists r'{\in}R'.\ p' \sqsubseteq r'$ and $\forall q'{\in}Q'\, \exists r'{\in}R'.\ q' \sqsubseteq r'$. Thus, $p_0 \vee q_0 \xrightarrow{i}\overset{\varepsilon}{\Longrightarrow} P' \cup Q'$ using Rule (IMust) and applying Def. 2;recall that $P \cap Q = \emptyset$.

(iii) Let $r \xrightarrow{\omega}_R R'$. By $p_0 \sqsubseteq r$ and $q_0 \sqsubseteq r$ we have P', Q' such that $p_0 \stackrel{\hat{\omega}}{\Longrightarrow}_P P'$, $q_0 \stackrel{\hat{\omega}}{\Longrightarrow}_Q Q'$ and $\forall p' \in P' \cup Q'\, \exists r' {\in} R'.\ p' \sqsubseteq r'$. Hence, $p_0 \vee q_0 \stackrel{\hat{\omega}}{\Longrightarrow} P' \cup Q'$ due to Rule (Must) and Def. 2.

(iv) Let $p_0 \vee q_0 \stackrel{i}{\dashrightarrow}_Q$. Then, w.l.o.g., we only need to consider $p_0 \stackrel{i}{\dashrightarrow}_P p'$, and because $p_0 \sqsubseteq r$ we have $r \stackrel{i}{\dashrightarrow}\stackrel{\varepsilon}{=\!\Rightarrow}_R r'$ for some r' satisfying $p' \sqsubseteq r'$.

(v) Let $p_0 \vee q_0 \stackrel{\omega}{\dashrightarrow}$. This is only possible for $\omega = \tau$. W.l.o.g. we only need to consider $p_0 \vee q_0 \stackrel{\tau}{\dashrightarrow} p_0$. This transition is matched with $r \stackrel{\varepsilon}{=\!\Rightarrow}_R r$ since $p_0 \sqsubseteq r$. $\square$

Corollary 42. *MIA refinement is compositional wrt. disjunction.*

6. Alphabet Extension

So far, MIA refinement is only defined on MIAs with the same alphabets. This is insufficient for supporting perspective-based specification, where an overall specification is conjunctively composed of smaller specifications, each addressing one 'perspective' (e.g., a single system requirement) and referring only to actions that are relevant to that perspective. Hence, it is useful to extend conjunction and thus MIA refinement to dissimilar alphabets in such a way that we can add new inputs and outputs in a refinement step. For this purpose we introduce alphabet extension as an operation on MIAs, similar to [12] for a *pessimistic* interface theory and also to *weak extension* in [10]. More precisely, we add may-loops for all new actions to each state, except to the universal state. Intuitively, the extended MIA can ignore all new actions while keeping control over the old ones. To express this, it is important to have, in particular, input may-transitions that determine how the MIA behaves subsequently. Such transitions were not available in the *optimistic* interface theory in [12]. Conjunction and also disjunction are now easily generalised by applying alphabet extension to the operands. The extended refinement preorder is compositional wrt. all our operators, except for the quotient where the situation is more difficult as we discuss below.

Definition 43 (Alphabet Extension & Refinement)**.** *Given a MIA* $(P, I, O, \longrightarrow, \dashrightarrow, p_0, e)$ *and disjoint action sets* I' *and* O' *satisfying* $I' \cap A = \emptyset = O' \cap A$ *for* $A =_{df} I \cup O$*, the* alphabet extension of P by I' and O' *is given by* $[P]_{I',O'} =_{df} (P, I \cup I', O \cup O', \longrightarrow, \dashrightarrow', p_0, e)$ *for* $\dashrightarrow' =_{df} \dashrightarrow \cup \{(p, a, p) \mid p \in P \setminus \{e\},\ a \in I' \cup O'\}$*. We often write* $[p]_{I',O'}$ *for* p *as state of* $[P]_{I',O'}$*, or conveniently* $[p]$ *in case* I'*,* O' *are understood from the context.*

For MIAs P *and* Q *with* $p \in P$*,* $q \in Q$*,* $I_P \supseteq I_Q$ *and* $O_P \supseteq O_Q$*, we define* $p \sqsubseteq' q$ *if* $p \sqsubseteq [q]_{I_P \setminus I_Q,\, O_P \setminus O_Q}$*. Since* $\sqsubseteq'$ *extends* $\sqsubseteq$ *to MIAs with different alphabets, we write* $\sqsubseteq$ *for* $\sqsubseteq'$ *and abbreviate* $[q]_{I_P \setminus I_Q,\, O_P \setminus O_Q}$ *by* $[q]_P$*; the same notations are used for* P *and* Q*.*

As an aside we remark that our alphabet extension is different to the one proposed by Ben-David et al. for Modal Transition Systems in [29], where unknown actions are treated as internal actions. Doing so has the consequence, however, that a state with an a-must-transition can be refined by a state that offers a b-must-transition followed by an a-must-transition, where b is a new action. In the context of interface theories, if a is an input, this is undesirable. If a is an output, the refinement is also not plausible for an input b since inputs are not locally controlled. However, for an output b, the approach of [29] could be considered for MIA, too.

It is easy to show that compositionality of parallel composition as in Thm. 15 is preserved by the extended refinement relation as long as alphabet extension does not yield new communications:

Theorem 44 (Compositionality of Parallel Composition)**.** *Let P_1, P_2, Q be MIAs such that Q and P_2 are composable and $P_1 \sqsubseteq Q$. Assume further that, for $I' =_{df} I_1 \setminus I_Q$ and $O' =_{df} O_1 \setminus O_Q$, we have $(I' \cup O') \cap A_2 = \emptyset$. Then:*

1. *P_1 and P_2 are composable.*
2. *If Q and P_2 are compatible, then so are P_1 and P_2 and $P_1 \parallel P_2 \sqsubseteq Q \parallel P_2$.*

Proof. It is easy to see that the MIAs $[Q]_{I',O'}$ and P_2 are composable due to $(I' \cup O') \cap A_2 = \emptyset$, which implies Claim 1. Furthermore, $[Q]_{I',O'} \otimes P_2$ is isomorphic to $[Q \otimes P_2]_{I',O'}$ via mapping $[q] \otimes p_2 \mapsto [q \otimes p_2]$. This is because of (PMay1) in the definition of $\otimes$, since we only add "fresh" may-transitions to each $q \in Q$. The mapping also respects errors: new may-transitions with label $o \in O'$ cannot create new errors since $o \notin I_2$, and no new $i \in I'$ has to have a must-transition since $i \notin O_2$. Thus, $[q_0]$ and p_{02} are compatible if q_0 and p_{02} are; moreover, $p_{01} \sqsubseteq [q_0]$. Now, the result follows from Thm. 15. $\Box$

It is obvious that new communications might result in an error and, therefore, must be disallowed. Technically, if $a \in (A_1 \setminus A_Q) \cap A_2$, then $P_1 \parallel P_2$ might have a new error if P_1 performs $a \in O_1$ or cannot perform $a \in I_1$.

It is also easy to see that the generalised $\sqsubseteq$ is a precongruence for hiding and restriction as well.

Proposition 45. *Let P and Q be MIAs such that $P \sqsubseteq Q$.*

1. *$P/L \sqsubseteq Q/L$, for any set L of actions with $L \cap I_P = \emptyset$.*
2. *$P \setminus L \sqsubseteq Q \setminus L$, for any set L of actions with $L \cap O_P = \emptyset$.*

Proof. We have $P \sqsubseteq [Q]_P$ and, due to Prop. 22, $P/L \sqsubseteq [Q]_P/L$ and $P \setminus L \sqsubseteq [Q]_P \setminus L$. First, $[Q]_P/L$ and $[Q/L]_{P/L}$ differ only by additional τ-loops in the former, arising from $o \in (O_P \setminus O_Q) \cap L$; hence, they are related by $\sqsupseteq\sqsubseteq$. Second, $[Q]_P \setminus L$ and $[Q \setminus L]_{P \setminus L}$ are identical. $\Box$

Similarly, $[P]_{\emptyset,O'}/O'$ and P differ only by an additional τ-loop at each state of the former; thus:

Proposition 46. *Let P be a MIA and $O' \cap O = \emptyset$. Then, $[P]_{\emptyset,O'}/O' \sqsupseteq\sqsubseteq P$.*

Now, we lift our conjunction operator to conjuncts with dissimilar alphabets:

Definition 47 (Lifting Conjunction)**.** *Let P, Q be MIAs, $p \in P$ and $q \in Q$ such that $I_P \cap O_Q = \emptyset = I_Q \cap O_P$. Then, $p \wedge' q =_{df} [p]_Q \wedge [q]_P$ and similarly for $P \wedge' Q$.*

We simply write $\wedge$ for $\wedge'$ in the following. To be able to lift our main result, Thm. 36, it is sufficient to establish that the alphabet extension operation is a homomorphism for conjunction. The proof of Thm. 49 below follows exactly the line of argument in [12].

Lemma 48. *Let P with $p \in P$ and Q with $q \in Q$ be MIAs with common alphabets. Consider the alphabet extensions by some I' and O'. Then:*

1. *p and q are consistent iff $[p]$ and $[q]$ are.*
2. *Given consistency, $[p \wedge q] \sqsupseteq\sqsubseteq [p] \wedge [q]$.*

Proof. For proving Claim 1, consider the mapping $\beta : (p,q) \mapsto ([p],[q])$, which is a bijection between $P\&Q$ and $[P]\&[Q]$. We have $(p,q) \in F_{P\&Q}$ due to $a \in A$ and (F1), (F2), (F3) or (F4) iff $([p],[q]) \in F_{[P]\&[Q]}$ due to $a \in A$ and (F1), (F2), (F3) or (F4). Observe that (F1), (F2), (F3) and (F4) never apply to $([p],[q])$ and $a \in I' \cup O'$, since there are no must-transitions labelled a. For the same reason, Rules (OMust1), (OMust2), (IMust1), (IMust2), (EMust1) and (EMust2) are never applicable for a and, thus, β is an isomorphism regarding must-transitions; hence, (F5) is applicable exactly in the corresponding cases according to β. Therefore, β is also a bijection between $F_{P\&Q}$ and $F_{[P]\&[Q]}$.

For Claim 2, we can regard β also as a bijection between $[P \wedge Q]$ and $[P]\wedge[Q]$, and establish each direction of $\sqsupseteq\sqsubseteq$ separately:

- *"$\sqsubseteq$":* We show that β is a MIA-refinement relation, for which we consider $[p \wedge q]$ and $[p] \wedge [q]$. Cond. (i) of Def. 4 is trivial. Conds. (ii) and (iii) are clear, because β is still an isomorphism on must-transitions. Regarding Conds. (iv) and (v), we only have to consider $\alpha \in I' \cup O'$ and $[p \wedge q] \stackrel{\alpha}{\dashrightarrow} [p \wedge q]$. This transition can be matched by the transition $[p]\wedge[q] \stackrel{\alpha}{\dashrightarrow} [p]\wedge[q]$, which exists by (IMay), (OMay), (EMay1) or (EMay2).
- *"$\sqsupseteq$":* We show that also β^{-1} is a MIA-refinement relation. Take $[p] \wedge [q]$ and $[p \wedge q]$; again, Conds. (i), (ii) and (iii) are clear. Thus, we only have to consider $\alpha \in I' \cup O'$ to establish Conds. (iv) and (v), so that $[p] \wedge [q] \stackrel{\alpha}{\dashrightarrow} r$ iff $r = [p'] \wedge [q']$ for $p \stackrel{\varepsilon}{\Longrightarrow} p'$ and $q \stackrel{\varepsilon}{\Longrightarrow} q'$. Such a transition can be matched by the transition $[p \wedge q] \stackrel{\alpha}{\dashrightarrow} [p \wedge q] \stackrel{\varepsilon}{\Longrightarrow} [p' \wedge q']$, where the weak may-transition exists by (May1), (May2), (OMay), (IMay), (EMay1) or (EMay2), or because $p = p'$ and $q = q'$. $\square$

Theorem 49 ($\wedge$ is And)**.** *Let P, Q and R be MIAs such that $I_P \cap O_Q = \emptyset = I_Q \cap O_P$, $I_R \supseteq I_P \cup I_Q$ and $O_R \supseteq O_P \cup O_Q$.*

1. *There exists such an R with $R \sqsubseteq P$ and $R \sqsubseteq Q$ iff $P \wedge Q$ is defined.*

2. *In case $P \wedge Q$ is defined: $R \sqsubseteq P$ and $R \sqsubseteq Q$ iff $R \sqsubseteq P \wedge Q$.*

Proof. Recall that we denote by $[\cdot]_P$ an extension with the additional actions of P, and similarly for Q and R. Also note that, in the context of this theorem, $[[p_0]_Q]_R = [p_0]_R$ and $[[q_0]_P]_R = [q_0]_R$.

Claim 1: If $r_0 \sqsubseteq [p_0]_R$ and $r_0 \sqsubseteq [q_0]_R$, then $[p_0]_R \wedge [q_0]_R$ is defined by Thm. 36. The latter conjunction equals $[[p_0]_Q]_R \wedge [[q_0]_P]_R$; hence, $[p_0]_Q \wedge [q_0]_P$ is defined by Lem. 48, and this conjunction is $p_0 \wedge q_0$ by definition. If $[p_0]_Q \wedge [q_0]_P$ is defined, there exists R with the common alphabets of $[P]_Q$ and $[Q]_P$ with $r_0 \sqsubseteq [p_0]_Q$ and $r_0 \sqsubseteq [q_0]_P$ by Thm. 36. For this R, we have $[p_0]_Q = [p_0]_R$ and $[q_0]_P = [q_0]_R$; thus, $r_0 \sqsubseteq p_0$ and $r_0 \sqsubseteq q_0$ by definition.

Claim 2: Let $p_0 \wedge q_0$ be defined. We reason as follows:

	$r_0 \sqsubseteq p_0$ and $r_0 \sqsubseteq q_0$	
iff	$r_0 \sqsubseteq [p_0]_R$ and $r_0 \sqsubseteq [q_0]_R$	(by definition)
iff	$r_0 \sqsubseteq [p_0]_R \wedge [q_0]_R$	(by Thm. 36)
iff	$r_0 \sqsubseteq [[p_0]_Q \wedge [q_0]_P]_R$	(by Lem. 48 and note above)
iff	$r_0 \sqsubseteq p_0 \wedge q_0$	(by Defs. 43 and 47) $\square$

The situation for disjunction under alphabet extension is analogous to the one above, but exploiting monotonicity of the alphabet extension operation wrt. $\sqsubseteq$.

Definition 50 (Lifting Disjunction)**.** *Let P, Q be MIAs, $p \in P$ and $q \in Q$ such that $I_P \cap O_Q = \emptyset = I_Q \cap O_P$. Then, $p \vee' q =_{df} [p]_Q \vee [q]_P$ and similarly for $P \vee' Q$. Once again, we simply write $\vee$ for $\vee'$.*

Lemma 51 (Monotonicity of $[\cdot]$)**.** *Let P with $p \in P$ and R with $r \in R$ be MIAs with common alphabets, as well as I' and O' be suitable action sets for extending them. Then, $p \sqsubseteq r$ iff $[p] \sqsubseteq [r]$.*

Proof. Since we only add may-loops with a fresh label a for the extension, it suffices to observe for direction "$\Longrightarrow$" and $p \sqsubseteq r$ that each may-transition $[p] \overset{a}{\dashrightarrow} [p]$ can be matched by $[r] \overset{a}{\dashrightarrow} [r]$, or $r = e_R$. $\square$

Theorem 52 ($\vee$ is Or)**.** *Let P, Q and R be MIAs such that $I_P \cap O_Q = \emptyset = I_Q \cap O_P$, $I_R \subseteq I_P \cup I_Q$ and $O_R \subseteq O_P \cup O_Q$. Then, $P \vee Q \sqsubseteq R$ iff $P \sqsubseteq R$ and $Q \sqsubseteq R$.*

Proof. The proof proceeds along the following chain of equivalences:

	$p_0 \vee q_0 \sqsubseteq r_0$	
iff	$[p_0]_Q \vee [q_0]_P \sqsubseteq [[r_0]_P]_Q$	(by definition)
iff	$[p_0]_Q \sqsubseteq [[r_0]_P]_Q$ and $[q_0]_P \sqsubseteq [[r_0]_P]_Q$	(by Thm. 41)
iff	$p_0 \sqsubseteq [r_0]_P$ and $q_0 \sqsubseteq [r_0]_Q$	(by Lem. 51)
iff	$p_0 \sqsubseteq r_0$ and $q_0 \sqsubseteq r_0$	(by definition) $\square$

We conclude this section by reconsidering our quotient operator. As discussed in Sec. 4.2, there is some freedom in choosing the input alphabet of the quotient $P /\!\!/ D$ of a specification P and a divisor D, namely $I_P \setminus I_D \subseteq I_{P /\!\!/ D} \subseteq$

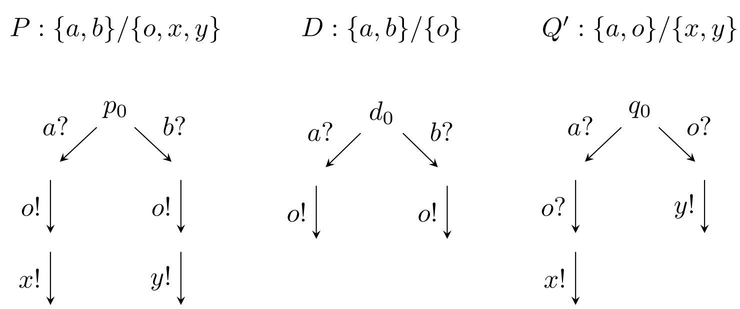

Figure 8: Complications of quotienting in the context of alphabet extension.

$I_P \cup O_D$. Since our extended refinement allows us to compare MIAs with different alphabets, one could aim for a generalisation of Thm. 30 where Q and $P /\!\!/ D$ may have different alphabets.

Because $Q \sqsubseteq P /\!\!/ D$, the quotient should have a minimal alphabet in this version, in contrast to our choice of $I_{P/\!\!/D} = I_P \cup O_D$. However, this leads to complications as one can see from the example in Fig. 8. A MIA Q satisfying $Q \,\|\, D \sqsubseteq P$ must have $O_Q = \{x, y\}$, but $I_Q = I_P \setminus I_D = \emptyset$ clearly does not suffice because Q is allowed to produce x or y only after o. Furthermore, Q must see a or b to distinguish between the branches. Solutions are possible, e.g., for $I_Q = \{a, o\}$ and $I_Q = \{b, o\}$; a solution Q' for $\{a, o\}$ is also shown in Fig. 8, where transitions to the universal state are not drawn for simplicity. It looks like there are several maximal solutions.

Note, however, that Thm. 30 in its present form still holds for our extended refinement preorder. This is important in practice where one would want for Q and D to be able to communicate via new internal actions, i.e., those that are hidden immediately after taking the parallel composition of Q and D. Since only outputs can be hidden, the new actions must form a set O' of outputs in $Q \,\|\, D$. Then, one proceeds by determining $Q =_{\mathrm{df}} [P]_{\emptyset,O'} /\!\!/ D$. Thm. 30 implies $Q \,\|\, D \sqsubseteq [P]_{\emptyset,O'}$, which in turn implies $(Q \,\|\, D)/O' \sqsubseteq P$ by Props. 22.1 and 46.

Another aspect of alphabet extension for quotienting is that we can generalise the problem by permitting D to have actions unknown to P. A straightforward generalisation of our approach in Sec. 4 would make these actions inputs for the quotient, but there can also be solutions to $Q \,\|\, D \sqsubseteq P$ where Q has some new inputs of D as outputs. We leave a further investigation of these aspects to future work.

7. Example

In this section we discuss an example, which demonstrates how MIA can be applied in practice. It exercises all important operations of MIA; it also uses nondeterminism which means that it cannot be modelled in MI [10]. The

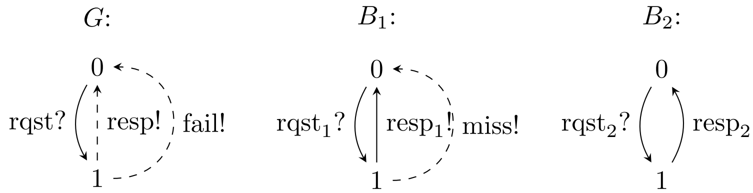

Figure 9: Global specification G, local cache B_1 and remote database B_2.

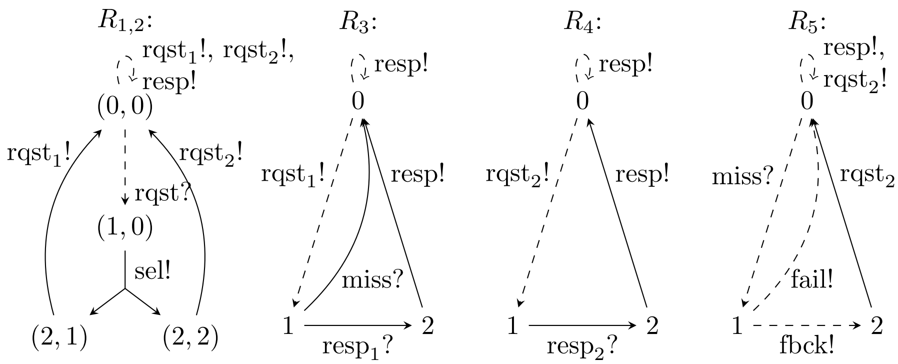

Figure 10: Front-end requirements $R_{1,2}$ (see also Fig. 7) and R_3, R_4, R_5.

example has been checked by a simple computer tool, which has been written by us and implements the MIA operations.

We consider a data server S that is composed of a front-end F and two already existing back-ends, a local cache B_1 and a remote database B_2. The server channels requests received by the front-end to (one of) the two back-ends. Based on a global specification G of S, we wish to develop the specification of F. The global specification and the back-end specifications are shown in Fig. 9.

Specification G defines the communication protocol with a client. The data server shall wait for a request and then may return a response or, alternatively, a failure message. Action rqst? is of must-modality because a data server makes no sense if it cannot accept a request. Actions resp! and fail! are of may-modality since refinements of G might at some stage decide to give only answer resp! or only fail!. The local cache B_1 also waits for a request and answers with a response; optionally, it may implement a cache miss after a request. The remote database B_2 is similar to the cache but without a miss. In both cases we have must-transitions for rqst_i? and resp_i!, so that the acceptance of inputs and issuing of answers is guaranteed.

We now develop the front-end specification F, which forwards a request to either cache B_1 or to database B_2. In case of the former and a cache miss, F may fall back to B_2. To this end, we assume the following requirements for F, which are specified in Fig. 10:

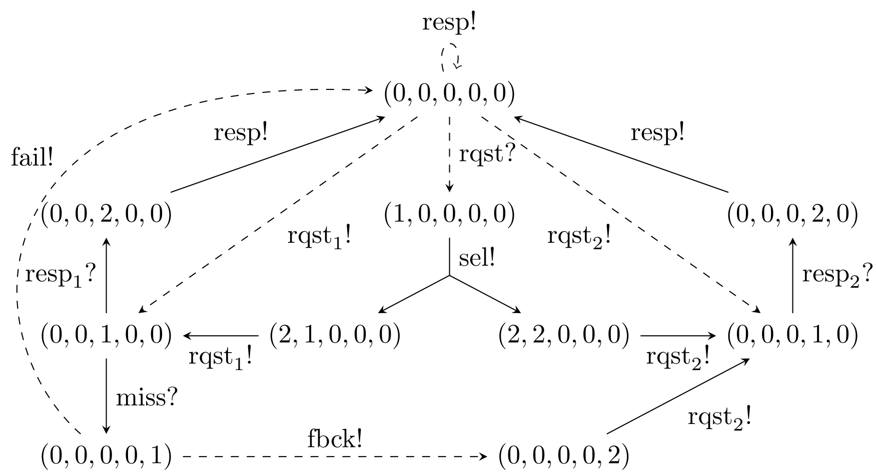

Figure 11: Conjunction of the front-end requirements, $R = R_{1,2} \wedge R_3 \wedge R_4 \wedge R_5$ with the alphabets $I = \{\text{rqst}, \text{resp}_1, \text{resp}_2, \text{miss}\}$, $O = \{\text{resp}, \text{rqst}_1, \text{rqst}_2, \text{sel}, \text{fbck}, \text{fail}\}$.

($R_{1,2}$) The front-end shall pass on a client's request to one of the back-ends.

(R_3) After forwarding a request to back-end B_1, the front-end shall wait for B_1's response and route it back to the client. Additional to the response, the front-end shall accept a cache miss when waiting for a response.

(R_4) After redirecting the request to back-end B_2, the front-end shall wait for B_2's response and route it back to the client.

(R_5) In case of a cache miss, the front-end may fall back to the database or fail.

Requirement $R_{1,2}$ is already discussed in Sec. 5 (cf. Fig. 7). Requirement R_3 states that, after forwarding a request to the cache ($\text{rqst}_1!$), the front-end must wait for a response ($\text{resp}_1?$) or a cache miss (miss?). In case of a response ($\text{resp}_1?$), the response has to be routed back to the client (resp!). Requirement R_4 is the corresponding requirement for the database back-end. Requirement R_5 specifies that, in case of a cache miss, the request can be redirected to the database back-end (fbck!) or the whole conversation may fail (fail!).

The conjunction $R =_{\text{df}} R_{1,2} \wedge R_3 \wedge R_4 \wedge R_5$ is shown in Fig. 11, where inconsistent and unreachable states are already pruned. Observe that one could simplify R by merging states $(0, 0, 0, 0, 2) \sqsupseteq\sqsubseteq (2, 2, 0, 0, 0)$.

All in all, the desired front-end specification F must guarantee that (i) the server S obeys the global specification, (ii) S is the parallel composition of the front-end and the two back-ends, and (iii) F satisfies all its requirements. Formally:

$$S \sqsubseteq G \qquad\qquad S = F \parallel B_1 \parallel B_2 \qquad\qquad F \sqsubseteq R$$

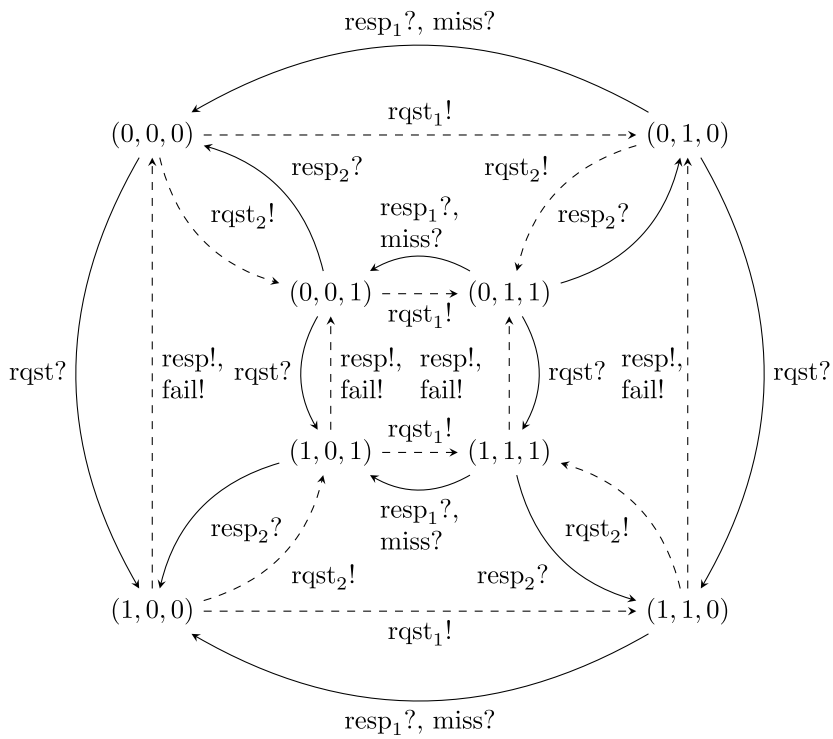

Figure 12: Upper bound U_F on F with the alphabets $I =_{\mathrm{df}} \{\mathrm{rqst}, \mathrm{resp}_1, \mathrm{resp}_2, \mathrm{miss}\}$ and $O =_{\mathrm{df}} \{\mathrm{resp}, \mathrm{rqst}_1, \mathrm{rqst}_2, \mathrm{fail}\}$.

Quotienting now gives us an upper bound U_F on F. To satisfy the alphabet requirements for quotienting, we first need to extend G's alphabet with the unknown actions $O' =_{\mathrm{df}} \{\mathrm{rqst}_1, \mathrm{rqst}_2, \mathrm{resp}_1, \mathrm{resp}_2, \mathrm{miss}\}$ of $B_1 \parallel B_2$; see the discussion at the end of Sec. 6 and observe that these actions are indeed outputs in the parallel composition of F with $B_1 \parallel B_2$. Now,

$$U_F = [G]_{\emptyset, O'} /\!\!/ (B_1 \parallel B_2)\,,$$

i.e., U_F (see Fig. 12) is the least specific interface that composes with the back-ends such that, after hiding of O', they together satisfy the global specification, as discussed in Sec. 6. Hence, overall:

$$(U_F \parallel B_1 \parallel B_2)/O' \sqsubseteq G.$$

Note that, in Fig. 12, we have omitted the universal state and its transitions, labelled resp_1, resp_2 and miss. These transitions do not play a role in $U_F \wedge R$ in the next step, because the only three transitions in R with these labels are solely combined with transitions actually shown in Fig. 12.

Thus, the front-end is specified as follows, because it also has to satisfy the

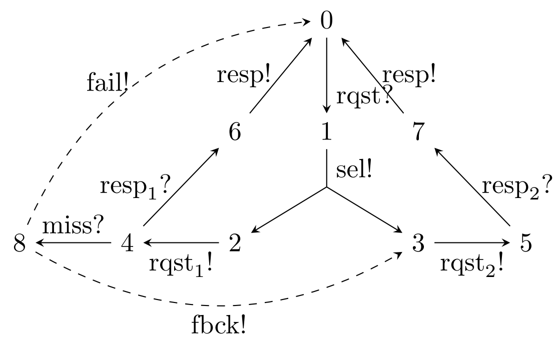

Figure 13: Final specification F of the front-end.

requirements given by R:

$$F =_{\mathrm{df}} U_F \wedge R$$

This specification leaves the implementor as much freedom as possible. It is shown in Fig. 13, where all unreachable and inconsistent states have already been removed.

8. Conclusions and Future Work

We presented an extension of Raclet et al.'s modal interface theory MI [10] to *nondeterministic* systems. To do so we resolved, for the first time properly, the conflict between unspecified inputs being allowed in interface theories derived from de Alfaro and Henzinger's Interface Automata [4] but forbidden in Modal Transition Systems [11]. To this end, we introduced a special universal state, which enabled us to achieve compositionality (in contrast to [8]) as well as associativity (in contrast to [10]) for parallel composition; crucially, this also enabled a more practical support of perspective-based specification when compared to [9, 12]. As another important contribution, we defined a quotienting operator that permits the decomposition of *nondeterministic* specifications and takes pruning in parallel composition into account (in contrast to [10]). In addition, we also introduced hiding and restriction for event scoping and disjunction as the dual to conjunction.

We are currently exploring the utility of MIA as a *behavioural type theory* for parallel programming languages. To this end we have enriched Google's Go language with such behavioural types, whereby type checking becomes refinement checking [30]. Our refinement checker is implemented via a translation to quantified boolean formulas into an SMT problem, along the lines of a similar translation for Modal Transition Systems [31].

Regarding further future work, we wish to explore the choice of alphabets for quotienting and relax the determinism requirement on divisors. We also intend to implement our interface theory in existing formal methods tools, such as

MICA (see `http://www.irisa.fr/s4/tools/mica/`), the MIO Workbench [5] or MoTraS [32].

9. References

[1] F. Bujtor, S. Fendrich, G. Lüttgen, W. Vogler, Nondeterministic modal interfaces, in: Theory and Practice of Computer Science (SOFSEM), Vol. 8939 of LNCS, Springer, 2015, pp. 152–163.

[2] S. S. Bauer, A. David, R. Hennicker, K. G. Larsen, A. Legay, U. Nyman, A. Wasowski, Moving from specifications to contracts in component-based design, in: Fundamental Approaches to Software Engineering (FASE), Vol. 7212 of LNCS, Springer, 2012, pp. 43–58.

[3] D. Beyer, A. Chakrabarti, T. A. Henzinger, S. A. Seshia, An application of web-service interfaces, in: Web Services (ICWS), IEEE, 2007, pp. 831–838.

[4] L. de Alfaro, T. A. Henzinger, Interface-based design, in: Engineering Theories of Software-Intensive Systems, Vol. 195 of NATO Science Series, Springer, 2005, pp. 83–104.

[5] S. S. Bauer, P. Mayer, A. Schroeder, R. Hennicker, On weak modal compatibility, refinement, and the MIO Workbench, in: Tools and Algorithms for the Construction and Analysis of Systems (TACAS), Vol. 6015 of LNCS, Springer, 2010, pp. 175–189.

[6] T. Chen, C. Chilton, B. Jonsson, M. Z. Kwiatkowska, A compositional specification theory for component behaviours, in: Programming Languages and Systems (ESOP), Vol. 7211 of LNCS, Springer, 2012, pp. 148–168.

[7] C. Chilton, An algebraic theory of componentised interaction, Ph.D. thesis, Oxford (2013).

[8] K. G. Larsen, U. Nyman, A. Wasowski, Modal I/O automata for interface and product line theories, in: Programming Languages and Systems (ESOP), Vol. 4421 of LNCS, Springer, 2007, pp. 64–79.

[9] G. Lüttgen, W. Vogler, Modal interface automata, Logical Methods in Computer Science (LMCS) 9 (3).

[10] J.-B. Raclet, E. Badouel, A. Benveniste, B. Caillaud, A. Legay, R. Passerone, A modal interface theory for component-based design, Fundamenta Informaticae 108 (1-2) (2011) 119–149.

[11] K. G. Larsen, Modal specifications, in: Automatic Verification Methods for Finite State Systems, Vol. 407 of LNCS, Springer, 1989, pp. 232–246.

[12] G. Lüttgen, W. Vogler, S. Fendrich, Richer interface automata with optimistic and pessimistic compatibility, Acta Informatica 52 (4-5) (2014) 305–336.

[13] C. A. R. Hoare, Communicating Sequential Processes, Prentice-Hall, 1985.

[14] R. Milner, Communication and concurrency, Prentice Hall, 1989.

[15] N. A. Lynch, Distributed Algorithms, Morgan Kaufmann, 1996.

[16] A. Benveniste, B. Caillaud, D. Nickovic, R. Passerone, J.-B. Raclet, P. Reinkemeier, A. Sangiovanni-Vincentelli, W. Damm, T. A. Henzinger, K. G. Larsen, Contracts for system design, Tech. Rep. 8147, INRIA (2012).

[17] K. G. Larsen, L. Xinxin, Equation solving using modal transition systems, in: Logic in Computer Science (LICS), IEEE, 1990, pp. 108–117.

[18] L. M. Alonso, R. Peña, Acceptance automata: A framework for specifying and verifying TCSP parallel systems, in: Parallel Architectures and Languages Europe (PARLE), Vol. 506 of LNCS, Springer, 1991, pp. 75–91.

[19] J.-B. Raclet, Residual for component specifications, in: Formal Aspects of Component Software (FACS), Electronic Notes in Theoretical Computer Science (ENTCS), Vol. 215, Elsevier, 2008, pp. 93–110.

[20] N. Beneš, B. Delahaye, U. Fahrenberg, J. Křetínský, A. Legay, Hennessy-Milner logic with greatest fixed points as a complete behavioural specification theory, in: Concurrency Theory (CONCUR), Vol. 8052 of LNCS, Springer, 2013, pp. 76–90.

[21] H. Fecher, H. Schmidt, Comparing disjunctive modal transition systems with an one-selecting variant, Logic and Algebraic Programming 77 (1-2) (2008) 20–39.

[22] H. Hüttel, K. G. Larsen, The use of static constructs in a modal process logic, in: Logic at Botik, Vol. 363 of LNCS, Springer, 1989, pp. 163–180.

[23] L. de Alfaro, T. A. Henzinger, Interface automata, in: Foundations of Software Engineering (FSE), ACM, 2001, pp. 109–120.

[24] F. Bujtor, W. Vogler, Error-pruning in interface automata, in: Theory and Practice of Computer Science (SOFSEM), Vol. 8327 of LNCS, Springer, 2014, pp. 162–173.

[25] D. L. Dill, Trace Theory for Automatic Hierarchical Verification of Speed-Independent Circuits, MIT-Press, 1989.

[26] R. De Nicola, R. Segala, A process algebraic view of input/output automata, Theoretical Computer Science 138 (2) (1995) 391–423.

[27] M. van der Bijl, A. Rensink, J. Tretmans, Compositional testing with ioco, in: Formal Approaches to Software Testing (FATES), Vol. 2931 of LNCS, Springer, 2004, pp. 86–100.

[28] C. Chilton, B. Jonsson, M. Kwiatkowska, An algebraic theory of interface automata, Tech. Rep. RR-13-02, Univ. Oxford, U.K. (2013).

[29] S. Ben-David, M. Chechik, S. Uchitel, Merging partial behaviour models with different vocabularies, in: Concurrency Theory (CONCUR), Vol. 8052 of LNCS, Springer, 2013, pp. 91–105.

[30] J. Gareis, Prototypical Integration of the Modal Interface Automata Theory in Google Go, Master's thesis, Univ. Bamberg, Germany (2015).

[31] N. Beneš, J. Kretínský, K. G. Larsen, M. H. Møller, S. Sickert, J. Srba, Refinement checking on parametric modal transition systems, Acta Informatica 52 (2-3) (2015) 269–297.

[32] J. Křetínský, S. Sickert, Motras: A tool for modal transition systems and their extensions, in: Automated Technology for Verification and Analysis (ATVA), Vol. 8172 of LNCS, Springer, 2013, pp. 487–491.