Electronic Communications of the EASST
Volume 66 (2013)

Proceedings of the Automated Verification of Critical Systems (AVoCS 2013)

Richer Interface Automata with Optimistic and Pessimistic Compatibility

Gerald Lüttgen and Walter Vogler

15 pages

Guest Editors: Steve Schneider, Helen Treharne
Managing Editors: Tiziana Margaria, Julia Padberg, Gabriele Taentzer
ECEASST Home Page: http://www.easst.org/eceasst/ ISSN 1863-2122

Richer Interface Automata with Optimistic and Pessimistic Compatibility*

Gerald Lüttgen[1] **and Walter Vogler**[2]

[1] gerald.luettgen@swt-bamberg.de
Software Technologies Group
University of Bamberg, Germany

[2] vogler@informatik.uni-augsburg.de
Institute for Computer Science
University of Augsburg, Germany

Abstract: Modal transition systems are a popular semantic underpinning of interface theories, such as Nyman et al.'s IOMTS and Bauer et al.'s MIO, which facilitate component-based reasoning of concurrent systems. Our interface theory MIA repaired a compositional flaw of IOMTS-refinement and introduced a conjunction operator. In this paper, we first modify MIA to properly deal with internal computations including internal must-transitions, which were largely ignored already in IOMTS. We then study a MIA variant that adopts MIO's pessimistic – rather than IOMTS' optimistic – view on component compatibility and define, for the first-time in a pessimistic, non-deterministic setting, conjunction and disjunction on interfaces. For the pessimistic MIA variant we also provide a mechanism for extending alphabets when refining interfaces, which is a desired feature in practice. We illustrate our advancements via a small example.

Keywords: Interface theory, modal transitions system, modal interface automata, optimistic and pessmimistic view, interface refinement, alphabet extension.

1 Introduction

Interface theories [BMSH10, LNW07, LV13, RBB+11] are a key technology for the component-based design of critical systems and are applied, e.g., for specifying web services [BCHS07] and software contracts [BDH+12]. Many interface theories are inspired by de Alfaro and Henzinger's *Interface Automata* (IA) [dH05] which employs transition systems with input and output actions and alternating simulation for refinement. It is distinguished from classic process algebras by its parallel composition operator: an interface cannot block an incoming input in any state but, if an input arrives unexpectedly, this is treated as error, i.e., as an incompatibility. IA suffers from the fact that outputs cannot be required since any interface may be implemented by a component that accepts all inputs and does not engage in any output, hence avoiding errors altogether. This is undesired in practice and has led researchers to base interface theories on *modal* transition

* This research was supported by the DFG (German Research Foundation, grants LU 1748/3-1 and VO 615/12-1).

systems (MTS) [Lar90]; these distinguish between must- and may-transitions and thus allow one to enforce outputs via output must-transitions.

In the light of errors that may arise when joining components in parallel, two schools on MTS-based interface theories have emerged, which treat compatibility either optimistically or pessimistically. The pessimistic school of Bauer et al. [BMSH10] only defines the composition of a restricted set of components; however, their MIO setting employs standard *modal refinement* as refinement preorder and standard weak transitions for abstracting from internal computation. In contrast, the optimistic school of Nyman et al. [LNW07] follows IA in that parallel composition is still defined in the presence of error states, if some concrete system environment may prohibit such states to be reached. Their IOMTS setting is equipped with a customized preorder, which allows one to compose a much larger set of components than in MIO. Fatally, IOMTS-refinement does not require the matching of internal must-transitions of implementations and is not at all permissive wrt. abstracting from internal computation. Our interface theory *Modal Interface Automata* (MIA) [LV13] adopts IOMTS-refinement while repairing a compositional flaw regarding IOMTS parallel composition. It also adds conjunction on interfaces, which is a key operator allowing engineers to specify a concurrent system from different perspectives.

This paper advances the state-of-the-art of both schools. Regarding the optimistic MIA setting, we first re-consider IOMTS-refinement so that it properly deals with internal computation including internal must-transitions (cf. Sec. 2). Along the way we also permit general, disjunctive must-transitions, thereby increasing expressiveness and enabling an intuitive definition of disjunction on interfaces. To the best of our knowledge, no existing work on disjunctive MTS considers weak transitions. We then study a pessimistic variant of MIA and define, for the first-time in a pessimistic, non-deterministic setting, conjunction and disjunction on interfaces (cf. Sec. 3). While Bauer [Bau12] and Raclet et al. [RBB+11] also investigated conjunction, they did so only for deterministic interfaces not containing internal computation. Extending alphabets is useful in practice, firstly, when composing partial specification interfaces to an overall interface conjunctively and, secondly, since implementors may decide to add extra features that are not covered by the specification interface (cf. [RBB+11]). We add such a mechanism for the pessimistic MIA version and discuss the problems that arise with alphabet extension in the optimistic version.

In summary, we achieve a richer interface theory than related work does. In MIA, one may specify non-deterministic behaviour, enforce outputs, express disjunctive must-transitions, abstract from internal computation, interpret compatibility optimistically or pessimistically, compose interfaces also conjunctively and disjunctively, and (in the pessimistic version) extend alphabets. A small example dealing with a communication protocol illustrates our advancements (cf. Sec. 3.4).

2 Modal Interface Automata: The Optimistic Setting

This section fixes a severe shortcoming of MIA [LV13] that it inherited from IOMTS [LNW07], namely that the refinement preorder ignores the matching of must-transitions labelled with the internal action τ. The MIA variant presented below also permits (in contrast to [LV13]) general disjunctive must-transitions, thus enabling a natural definition of disjunction on interfaces.

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

(i) P is the set of states,

(ii) $A =_{\mathrm{df}} I \cup O$ with $I \cap O = \emptyset$ is the alphabet consisting of disjoint inputs and outputs, resp., and not containing the special, silent action τ,

(iii) $\longrightarrow \subseteq P \times (A \cup \{\tau\}) \times (\mathscr{P}_{fin}(P) \setminus \emptyset)$ is the *must-transition* relation (with $\mathscr{P}_{fin}(P)$ being the set of finite subsets of P),

(iv) $\dashrightarrow \subseteq P \times (A \cup \{\tau\}) \times P$ is the *may-transition* relation,

such that the following conditions hold for all $i \in I$ and $\alpha \in A \cup \{\tau\}$:

(a) $p \xrightarrow{i} P'$ and $p \xrightarrow{i} P''$ implies $P' = P''$ *(input determinism)*,

(b) $p \overset{i}{\dashrightarrow} p'$ implies $\exists P'.\, p \xrightarrow{i} P'$ and $p' \in P'$ *(input must)*,

(c) $p \xrightarrow{\alpha} P'$ implies $\forall p' \in P'.\, p \overset{\alpha}{\dashrightarrow} p'$ *(syntactic consistency)*.

Conds. (a)–(c) are adapted from the corresponding definition in [LV13]. Input determinism is required for the MIA-refinement preorder (see below) to be a precongruence for parallel composition and conjunction; this condition is already imposed by IA, but note that, here, an input must-transition is disjunctive, thus allowing nondeterminism within a transition. The input-must condition is natural in the presence of IA-inspired parallel composition: a may-input in an interface specification may simply be left out by a refining implementation, and thus increase the potential for errors rather than decrease it. Finally, syntactic consistency is natural and inherited from modal transition systems [Lar90].

In the sequel, we identify a MIA $(P, I, O, \longrightarrow, \dashrightarrow)$ 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., write $A = I/O$ when highlighting inputs I and outputs O in an alphabet A, and define $\hat{a} =_{\mathrm{df}} a$ and $\hat{\tau} =_{\mathrm{df}} \varepsilon$ (the empty word). In figures, we often refer to an action a as $a?$, if $a \in I$, and as $a!$, if $a \in O$, and omit the label of τ-transitions. Must-transitions (may-transitions) are drawn using solid, possibly splitting arrows (dashed arrows); any depicted must-transition also implicitly represents the resp. may-transition(s).

We now define *weak* must- and may-transition relations that abstract from transitions labelled by τ, as will be needed for MIA-refinement. This is the first definition of this kind, which covers disjunctive must-transitions; it is quite subtle as can be seen in Lemma 1 and Fig. 2 below.

Definition 2 (Weak Transition Relations) *Weak* must- and *weak* may-transition relations $\Longrightarrow$ and $=\!=\!\!\!\rightarrow$, resp., are defined as the smallest relations satisfying $p \overset{\varepsilon}{\Longrightarrow} \{p\}$, $p \overset{\varepsilon}{=\!=\!\!\!\rightarrow} p$ and the following conditions, where $\hat{\omega} \in O \cup \{\varepsilon\}$:

(a) $p \overset{\hat{\omega}}{\Longrightarrow} P'$, $p' \in P'$ and $p' \xrightarrow{\tau} P''$ implies $p \overset{\hat{\omega}}{\Longrightarrow} (P' \setminus \{p'\}) \cup P''$,

(b) $p \overset{\varepsilon}{\Longrightarrow} P' = \{p_1, \ldots, p_n\}$ and $\forall j.\, p_j \xrightarrow{o} P_j$, implies $p \overset{o}{\Longrightarrow} \bigcup_{j=1}^{n} P_j$,

(c) $p \overset{\varepsilon}{=\!=\!\twoheadrightarrow} p'' \overset{\tau}{\dashrightarrow} p'$ implies $p \overset{\varepsilon}{=\!=\!\twoheadrightarrow} p'$,

(d) $p \overset{\varepsilon}{=\!=\!\twoheadrightarrow} p'' \overset{\omega}{\dashrightarrow} p''' \overset{\varepsilon}{=\!=\!\twoheadrightarrow} p'$ implies $p \overset{\omega}{=\!=\!\twoheadrightarrow} p'$.

Our refinement relation is adapted from [LNW07, LV13] and called *MIA-refinement*:

Definition 3 (MIA-Refinement) Let P, Q be MIAs with common input and output alphabets. Relation $\mathscr{R} \subseteq P \times Q$ is a *MIA-refinement relation* if for all $(p,q) \in \mathscr{R}$:

(i) $q \xrightarrow{i} Q'$ implies $\exists P'.\, p \xrightarrow{i} P'$ and $\forall p' \in P'\, \exists q' \in Q'.\, (p',q') \in \mathscr{R}$,

(ii) $q \xrightarrow{\omega} Q'$ implies $\exists P'.\, p \overset{\hat{\omega}}{\Longrightarrow} P'$ and $\forall p' \in P'\, \exists q' \in Q'.\, (p',q') \in \mathscr{R}$,

(iii) $p \overset{\omega}{\dashrightarrow} p'$ implies $\exists q'.\, q \overset{\hat{\omega}}{=\!=\!\twoheadrightarrow} q'$ and $(p',q') \in \mathscr{R}$.

We write $p \sqsubseteq_{\smile} q$ and say that p *MIA-refines* q if there exists a MIA-refinement relation $\mathscr{R}$ such that $(p,q) \in \mathscr{R}$.

It is easy to see that $\sqsubseteq_{\smile}$ is the largest MIA-refinement relation and a preorder, i.e., it is reflexive and transitive. The key difference to [LV13] is that our revised definition of MIA above also allows τ-must-transitions that must be matched (Cond. (ii), for $\omega = \tau$). The reason why input must-transitions must be matched directly and not via a weak transition is due to the notion of MIA parallel composition, which we adopt from IA [dH05] and explain below. The same comment applies to the fact that our relation is insensitive to the refining MIA adding input-transitions, since input may-transitions are not considered in Cond. (iv).

2.1 Parallel Composition

We define a parallel composition operator $|$ on MIA in analogy to IA [dH05, LNW07] in two stages: first a standard product $\otimes$ between two MIAs is introduced, where common actions are synchronized and hidden. Then, error states are identified, and all states are pruned from which reaching an error state is unavoidable in some implementation.

Definition 4 (Parallel Product) MIAs P_1, P_2 are *composable* if $A_1 \cap A_2 = (I_1 \cap O_2) \cup (O_1 \cap I_2)$. For such MIAs we define the *product* $P_1 \otimes P_2 = (P_1 \times P_2,\, I, O,\, \longrightarrow,\, \dashrightarrow)$, where $I = (I_1 \cup I_2) \setminus (O_1 \cup O_2)$ and $O = (O_1 \cup O_2) \setminus (I_1 \cup I_2)$ and where $\longrightarrow$ and $\dashrightarrow$ are defined as follows:

(Must1)	$(p_1,p_2) \xrightarrow{\alpha} P_1' \times \{p_2\}$	if	$p_1 \xrightarrow{\alpha} P_1'$ and $\alpha \notin A_2$
(Must2)	$(p_1,p_2) \xrightarrow{\alpha} \{p_1\} \times P_2'$	if	$p_2 \xrightarrow{\alpha} P_2'$ and $\alpha \notin A_1$
(Must3)	$(p_1,p_2) \xrightarrow{\tau} P_1' \times P_2'$	if	$p_1 \xrightarrow{a} P_1'$ and $p_2 \xrightarrow{a} P_2'$ for some a
(May1)	$(p_1,p_2) \overset{\alpha}{\dashrightarrow} (p_1',p_2)$	if	$p_1 \overset{\alpha}{\dashrightarrow} p_1'$ and $\alpha \notin A_2$
(May2)	$(p_1,p_2) \overset{\alpha}{\dashrightarrow} (p_1,p_2')$	if	$p_2 \overset{\alpha}{\dashrightarrow} p_2'$ and $\alpha \notin A_1$
(May3)	$(p_1,p_2) \overset{\tau}{\dashrightarrow} (p_1',p_2')$	if	$p_1 \overset{a}{\dashrightarrow} p_1'$ and $p_2 \overset{a}{\dashrightarrow} p_2'$ for some a.

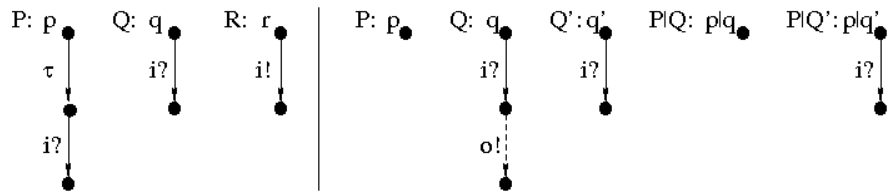

Figure 1: Necessity of matching input-transitions strongly (left) and of ignoring input may-transitions when matching (right).

The difference to the version of MIA in [LV13] is that we now have τ-must-transitions; in particular, this has led us to introduce Rule (Must3).

Definition 5 (Parallel Composition) Given a parallel product $P_1 \otimes P_2$, a state (p_1, p_2) is an *error state* 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 $p_1 \overset{a}{\not\longrightarrow}$. We define the set $E \subseteq P_1 \times P_2$ of *incompatible* states as the least set such that $(p_1, p_2) \in E$ if (i) (p_1, p_2) is an error state or (ii) $(p_1, p_2) \overset{\omega}{\dashrightarrow} (p_1', p_2')$ and $(p_1', p_2') \in E$.

The *parallel composition* $P_1|P_2$ of P_1 and P_2 is now obtained from $P_1 \otimes P_2$ by *pruning*, namely removing all states in E and every transition that involves such states as its source, its target or one of its targets; all may-transitions underlying a removed must-transition are deleted, too. If $(p_1, p_2) \in P_1|P_2$, we write $p_1|p_2$ and call p_1 and p_2 *compatible*.

It is easy to see that parallel products and parallel compositions are well-defined MIAs and that the parallel composition operator is commutative and associative. In addition and as we will show below, MIA-refinement is compositional wrt. parallel composition, i.e., $\sqsubseteq$ is a precongruence. It is this desired property that requires us in Def. 3 to match input must-transitions strongly and to ignore input may-transitions when matching. To see the former, consider Fig. 1 (left) with input/output alphabets $A_P =_{\text{df}} A_Q =_{\text{df}} \{i\}/\emptyset$ and $A_R =_{\text{df}} \emptyset/\{i\}$. Now, p should not refine q since q and r are compatible while p and r are not (because (p,r) is an error). Therefore, one must not be able to match a transition $\overset{i}{\longrightarrow}$ by a transition sequence $(\overset{\tau}{\longrightarrow})^+ \overset{i}{\longrightarrow}$, unless the notion of error state originating from Interface Automata [dH05] is changed, as is done in [BMSH10].

To see the latter, observe that prescribing the matching of input may-transitions as in Def. 3(iv) for output may-transitions would lead to a compositionality bug. For example, for the MIAs in Fig. 1 (right) with alphabets $A_P =_{\text{df}} \{o\}/\emptyset$ and $A_Q =_{\text{df}} A_{Q'} =_{\text{df}} \{i\}/\{o\}$ we would have $q' \sqsubseteq q$ but $p|q' \not\sqsubseteq p|q$ would fail.

Theorem 1 (Compositionality of Parallel Composition) *Let P_1, P_2, Q be MIAs with $p_1 \in P_1$, $p_2 \in P_2$, $q \in Q$ and $p_1 \sqsubseteq q$. Assume that Q and P_2 are composable; then:*

(a) *P_1 and P_2 are composable.*

(b) *If q and p_2 are compatible, then so are p_1 and p_2 and $p_1|p_2 \sqsubseteq q|p_2$.*

The proof of this result requires a couple of auxiliary properties regarding the preservation of

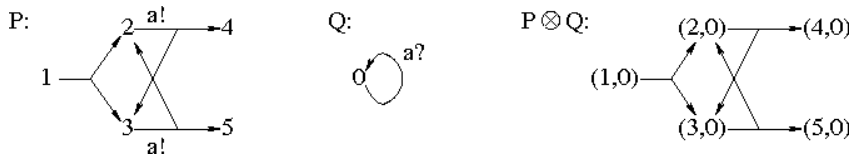

Figure 2: Example showing that set R in Lemma 1 is not always the full set $P' \times Q'$.

composability and consistency under refinement, respectively, as well as the following property of weak must-transitions:

Lemma 1 (Weak Must-Transitions) *Let P, Q be composable MIAs. If $p \stackrel{a}{\Longrightarrow}_P P'$ and $q \stackrel{a}{\longrightarrow}_Q Q'$ for some action $a \in (O_P \cap I_Q) \cup (I_P \cap O_Q)$, then $(p,q) \stackrel{\varepsilon}{\Longrightarrow} R$ in $P \otimes Q$ with $R \subseteq P' \times Q'$.*

Fig. 2 shows that, in general, $R \neq P' \times Q'$; here, $1 \stackrel{a!}{\Longrightarrow}_P \{2,3,4,5\}$ and $0 \stackrel{a?}{\Longrightarrow}_Q 0$, but not $1|0 \stackrel{\varepsilon}{\Longrightarrow} \{2,3,4,5\} \times \{0\}$. The sets R with maximal cardinality satisfying $1|0 \stackrel{\varepsilon}{\Longrightarrow} R$ are $\{2,4,5\} \times \{0\}$ and $\{3,4,5\} \times \{0\}$.

2.2 Conjunction & Disjunction

Conjunction on MIA will be defined in two stages, similarly to parallel composition. State pairs can be logically inconsistent due to unsatisfiable must-transitions and are then removed incrementally in the second stage.

Definition 6 (Conjunctive Product) Let $(P,I,O,\longrightarrow_P,\dashrightarrow_P)$ and $(Q,I,O,\longrightarrow_Q,\dashrightarrow_Q)$ be MIAs with common input and output alphabets and disjoint state sets. The conjunctive product $P\&Q =_{\text{df}} ((P \times Q) \cup P \cup Q, I, O, \longrightarrow, \dashrightarrow)$ inherits the transitions of P and Q and has additional transitions as follows:

(OMust1)	$(p,q) \stackrel{\omega}{\longrightarrow} \{(p',q') \mid p' \in P', q \stackrel{\hat{\omega}}{=\!=\!\!\rightarrow}_Q q'\}$	if	$p \stackrel{\omega}{\longrightarrow}_P P'$ and $q \stackrel{\hat{\omega}}{=\!=\!\!\rightarrow}_Q$
(OMust2)	$(p,q) \stackrel{\omega}{\longrightarrow} \{(p',q') \mid p \stackrel{\hat{\omega}}{=\!=\!\!\rightarrow}_P p', q' \in Q'\}$	if	$p \stackrel{\hat{\omega}}{=\!=\!\!\rightarrow}_P$ and $q \stackrel{\omega}{\longrightarrow}_Q Q'$
(IMust1)	$(p,q) \stackrel{i}{\longrightarrow} P'$	if	$p \stackrel{i}{\longrightarrow}_P P'$ and $q \not\stackrel{i}{\longrightarrow}_Q$
(IMust2)	$(p,q) \stackrel{i}{\longrightarrow} Q'$	if	$p \not\stackrel{i}{\longrightarrow}_P$ and $q \stackrel{i}{\longrightarrow}_Q Q'$
(IMust3)	$(p,q) \stackrel{i}{\longrightarrow} P' \times Q'$	if	$p \stackrel{i}{\longrightarrow}_P P'$ and $q \stackrel{i}{\longrightarrow}_Q Q'$
(May1)	$(p,q) \stackrel{\tau}{\dashrightarrow} (p',q)$	if	$p \stackrel{\tau}{=\!=\!\!\rightarrow}_P p'$
(May2)	$(p,q) \stackrel{\tau}{\dashrightarrow} (p,q')$	if	$q \stackrel{\tau}{=\!=\!\!\rightarrow}_Q q'$
(May3)	$(p,q) \stackrel{\omega}{\dashrightarrow} (p',q')$	if	$p \stackrel{\omega}{=\!=\!\!\rightarrow}_P p'$ and $q \stackrel{\omega}{=\!=\!\!\rightarrow}_Q q'$
(IMay1)	$(p,q) \stackrel{i}{\dashrightarrow} p'$	if	$p \stackrel{i}{\dashrightarrow}_P p'$ and $q \not\stackrel{j}{\dashrightarrow}_Q$
(IMay2)	$(p,q) \stackrel{i}{\dashrightarrow} q'$	if	$p \not\stackrel{j}{\dashrightarrow}_P$ and $q \stackrel{i}{\dashrightarrow}_Q q'$
(IMay3)	$(p,q) \stackrel{i}{\dashrightarrow} (p',q')$	if	$p \stackrel{i}{\dashrightarrow}_P p'$ and $q \stackrel{i}{\dashrightarrow}_Q q'$

Observe that the conjunctive product is inherently different from the parallel product, as can be

seen from some 'unusual' rules that define single transitions on the basis of weak transitions (Rules (OMust) and (May)) and synchronize on τ-transitions (Rule (May3)). These will be justified by Thm. 2 below. As an aside, note that we assume in the (OMust) rules and in similar cases below, that the target set of the defined transition is finite. If one wishes to deal with infinite target sets in MIA, one has to modify the definition of $\stackrel{\varepsilon}{\Longrightarrow}$ by allowing the simultaneous replacement of several p' by suitable P' in Def. 2(a); this would make the latter definition more complicated and Lemma 1 superfluous.

We now define a conjunction operator on MIAs that have the same input and output alphabets; relaxing this requirement will be discussed below.

Definition 7 (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:

(F1)	$p \stackrel{o}{\longrightarrow}_P$ and $q \not\stackrel{\rho}{=\!\!\!\Rightarrow}_Q$	implies	$(p,q) \in F$
(F2)	$p \not\stackrel{\rho}{=\!\!\!\Rightarrow}_P$ and $q \stackrel{o}{\longrightarrow}_Q$	implies	$(p,q) \in F$
(F3)	$(p,q) \stackrel{\alpha}{\longrightarrow} R'$ and $R' \subseteq F$	implies	$(p,q) \in F$

The conjunction $P \wedge Q$ of MIAs P,Q with common input and output alphabets is obtained 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.

Operator $\wedge$ indeed defines conjunction on MIA, i.e., $\wedge$ is the greatest lower bound wrt. $\sqsubseteq$:

Theorem 2 ($\wedge$ is And) *Let P and Q be MIAs with the same alphabets and disjoint state sets. We have (i) ($\exists$ MIA R and $r \in R.\, r \sqsubseteq p$ and $r \sqsubseteq q$) iff $p \wedge q$ is defined. Further, in case $p \wedge q$ is defined and for any MIA R and $r \in R$: (ii) $r \sqsubseteq p$ and $r \sqsubseteq q$ iff $r \sqsubseteq p \wedge q$. In both statements, R is supposed to have the same alphabets as P and Q.*

The theorem's first part reflects the intuition that specifications p and q are logically inconsistent if they do not have a common implementation; formally, $p \wedge q$ is undefined in this case. Its proof demands us to reason about inconsistent states, for which we resort to a notion of witness:

Definition 8 (Witness) A *witness* W of $P\&Q$ is a subset of $(P \times Q) \cup P \cup Q$ such that the following conditions hold for all $(p,q) \in W$:

(W1)	$p \stackrel{o}{\longrightarrow}_P$	implies	$q \stackrel{o}{=\!\!=\!\!\Rightarrow}_Q$
(W2)	$q \stackrel{o}{\longrightarrow}_Q$	implies	$p \stackrel{o}{=\!\!=\!\!\Rightarrow}_P$
(W3)	$(p,q) \stackrel{\alpha}{\longrightarrow} R'$	implies	$R' \cap W \neq \emptyset$

Lemma 2 (Concrete Witness) *Let $P\&Q$ be a conjunctive product of MIAs. Then, for any witness W of $P\&Q$, we have (i) $F \cap W = \emptyset$. Moreover, (ii) the set $W =_{df} \{(p,q) \in P \times Q \mid \exists$ MIA R and $r \in R.\, r \sqsubseteq p$ and $r \sqsubseteq q\} \cup P \cup Q$ is a witness of $P\&Q$.*

Statement (ii) above is now the key for proving Thm. 2. As a corollary to this theorem, one may

obtain compositionality of MIA-refinement wrt. conjunction:

Corollary 1 *If $p \sqsubseteq_{\smile} q$ and $p \wedge r$ defined, then $q \wedge r$ defined and $p \wedge r \sqsubseteq_{\smile} q \wedge r$.*

Note that one cannot expect that definedness of $q \wedge r$ implies that of $p \wedge r$, because specializing q to p might introduce an inconsistency.

We now turn our attention to defining the dual disjunction operator $\vee$ on MIA, which expresses the least upper bound property wrt. $\sqsubseteq_{\smile}$. The definition of disjunction may make use of the disjunctive must-transitions relation also for inputs and the internal action τ:

Definition 9 (Disjunction) Let $(P, I, O, \longrightarrow_P, \dashrightarrow_P)$ and $(Q, I, O, \longrightarrow_Q, \dashrightarrow_Q)$ be MIAs with common input and output alphabets and disjoint state sets. The disjunction $P \vee Q$ is defined by $(\{p \vee q \mid p \in P, q \in Q\} \cup P \cup Q, I, O, \longrightarrow, \dashrightarrow)$, 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)	$p \vee q \xrightarrow{\tau} \{p, q\}$	
(IMust)	$p \vee q \xrightarrow{i} P' \cup Q'$	if $p \xrightarrow{i}_P P'$ and $q \xrightarrow{i}_Q Q'$
(May)	$p \vee q \overset{\tau}{\dashrightarrow} p$, $p \vee q \overset{\tau}{\dashrightarrow} q$	
(May1)	$p \vee q \overset{i}{\dashrightarrow} p'$	if $p \overset{i}{\dashrightarrow}_P p'$ and $\exists q'.\, q \overset{i}{\dashrightarrow}_Q q'$
(May2)	$p \vee q \overset{i}{\dashrightarrow} q'$	if $q \overset{i}{\dashrightarrow}_Q q'$ and $\exists p'.\, p \overset{i}{\dashrightarrow}_P p'$

The idea behind the operational reading of $\vee$ is very intuitive since $p \vee q \xrightarrow{\tau} \{p, q\}$ naturally describes disjunctive behaviour. The only subtle point is that must-inputs must be matched directly, which justifies Rule (IMust) above. We now have the following desired theorem and corollary:

Theorem 3 ($\vee$ is Or) *Let P, Q and R be MIAs with common alphabets, disjoint state sets and states p, q, r, resp. Then, $p \vee q \sqsubseteq_{\smile} r$ iff $p \sqsubseteq_{\smile} r$ and $q \sqsubseteq_{\smile} r$.*

Corollary 2 *MIA-refinement is compositional wrt. disjunction.*

The above conjunction and disjunction operators are only defined on MIAs with the same alphabets. In practice, one would wish to also be able to apply these operators to MIAs that specify different aspects of the system under study and, thus, have different alphabets (cf. `Sender` and `Resetter` in Sec. 3.4). One way to deal with this situation is to extend the alphabets of the conjuncts in some $P \wedge Q$ to a common alphabet by adding a may-loop $p \overset{a}{\dashrightarrow} p$ to all states $p \in P$ and for all actions $a \in A_Q \setminus A_P$, and similarly for Q; such loops would express that P, Q behave neutral regarding actions that are not in their resp. alphabets. However, there is a problem with this idea in the context of MIA refinement, which we inherited from the IOMTS framework [LNW07].

To see this problem, consider the MIAs P, Q depicted in Fig. 3 with input/output alphabets $\emptyset/\{o, o'\}$ and resp. $\{i\}/\emptyset$, as well as MIAs R_1, R_2 and R_3 with alphabets $\{i\}/\{o, o'\}$. Intuitively, r_1 and r_2 should refine $p \wedge q$, while r_3 should not. This is because (i) p morally has an i-may-loop and q allows input i, and (ii) p enforces one output o and prohibits o' independent of any i. However, there is no MIA R with alphabets $\{i\}/\{o, o'\}$ and some $r \in R$ which has these properties of $p \wedge q$: if r_2 refines r, then so does r_3. The problem's source lies in the fact

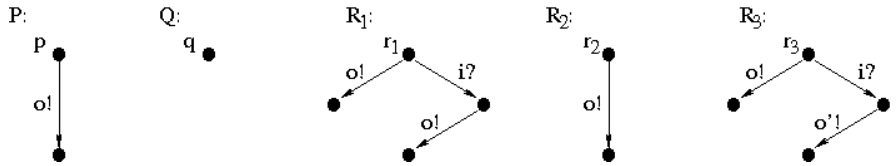

Figure 3: Alphabet extension and conjunction in the optimistic setting.

that, as shown in Fig. 1 (right), any reasonable refinement relation on MIA must allow additional inputs (with arbitrary subsequent behaviour) in implementations, if compositionality for parallel composition as in IA [dH05] should hold.

One could try to generalise Def. 3 such that the refining P may have additional inputs (and outputs) and decree that

$$p \xrightarrow{i} P' \text{ with } i \in I_P \setminus I_Q \text{ implies } \forall p' \in P'.\,(p', q) \in \mathscr{R},$$

as we did in a prior version of this paper that is included in the pre-proceedings of AVoCS'13. But the example above shows that the resulting refinement notion would not be transitive, since then $r_3 \sqsubseteq r_2 \sqsubseteq p$ but $r_3 \not\sqsubseteq p$. Our special treatment of inputs implies that may-inputs make no sense, because inputs are always implicitly allowed. Thus, one cannot formalize the decision not to implement some input in some state. As we will show in the next section, the *pessimistic* approach as, e.g., in [BMSH10] avoids this problem.

3 The Pessimistic Setting

Orthogonal to the 'optimistic' school on interface theories, comprising IA [dH05], IOMTS [LNW07] and the above MIA approaches, there is the school of Bauer et al. who has adopted a *pessimistic* view of compatibility in the presence of errors; see, e.g., [BMSH10]. Their interface theory, called MIO, also roots in Larsen's modal transition systems [Lar90] and allows may-inputs, but it defines parallel composition for much fewer interfaces when compared to optimistic approaches.

In our opinion, intuition for the pessimistic setting is weak since it distinguishes a state p where an input i is absent, from the situation where an i-transition leads to an error state; in both cases, an error is reached iff the environment provides input i. However, the pessimistic setting has technical advantages as we will see below. We will therefore re-develop our MIA theory for such a *pessimistic* setting, to which we will primarily contribute conjunction and disjunction operators and also *disjunctive* must-transitions. For completeness note that conjunction was defined by Bauer for a pessimistic interface theory in [Bau12]; however, he considered deterministic interfaces only and no internal actions.

Definition 10 (Relaxed MIA) A *Relaxed Modal Interface Automaton* (Relaxed MIA) is a tuple $(P, I, O, \longrightarrow, \dashrightarrow)$ as in Def. 1, but which is only required to satisfy *syntactic consistency*.

In the context of the pessimistic setting, it turns out that *input determinism* and *input must* (Conds. (a) and (b) of Def. 1) will not be necessary. We thus eliminate these conditions from

MIA and call the resulting automata *Relaxed MIAs*. In analogy to Def. 2 we now define weak transitions for Relaxed MIA and, for convenience, overload the according transition symbols:

Definition 11 (Relaxed Weak Transition Relations) The *relaxed weak must-transition relation* $\Longrightarrow$ and *relaxed weak may-transition relation* $\dashrightarrow$ are defined identically to the weak must- and may-transition relations in Def. 2, but replacing ω by α, $\hat{\omega}$ by $\hat{\alpha}$, and o by a. For input actions, we additionally define a restricted weak must-transition that only allows trailing τ-actions as follows:

(e) $p \xrightarrow{i} P'$ implies $p \overset{i}{\Longrightarrow} P'$,

(f) $p \overset{i}{\Longrightarrow} P'$, $p' \in P'$ and $p' \xrightarrow{\tau} P''$ implies $p \overset{i}{\Longrightarrow} (P' \setminus \{p'\}) \cup P''$,

Observe that $p \overset{i}{\Longrightarrow} P'$ implies $p \overset{i}{\Longrightarrow} P'$, which will be used in the sequel.

Since may-inputs are available in the pessimistic setting, extending the alphabets of interfaces can be defined via an according operation, as we will see below (Def. 15). Therefore, we first consider refinement and operators for Relaxed MIAs with the same input and output alphabets. The corresponding notions for Relaxed MIAs with dissimilar alphabets will then be defined on the basis of the existing ones and the alphabet extension operator.

Definition 12 (Modal Refinement on Relaxed MIA) Let P, Q be Relaxed MIAs with the same input and output alphabets. $\mathscr{R} \subseteq P \times Q$ is a *modal refinement relation* if for all $(p,q) \in \mathscr{R}$:

(i) $q \xrightarrow{i} Q'$ implies $\exists P'.\, p \overset{i}{\Longrightarrow} P'$ and $\forall p' \in P' \, \exists q' \in Q'.\, (p',q') \in \mathscr{R}$,

(ii) $q \xrightarrow{\omega} Q'$ implies $\exists P'.\, p \overset{\hat{\omega}}{\Longrightarrow} P'$ and $\forall p' \in P' \, \exists q' \in Q'.\, (p',q') \in \mathscr{R}$,

(iii) $p \overset{\alpha}{\dashrightarrow} p'$ implies $\exists q'.\, q \overset{\hat{\alpha}}{\dashrightarrow} q'$ and $(p',q') \in \mathscr{R}$.

We write $p \mathrel{\overset{\frown}{\sqsubseteq}} q$ and say that p *modal-refines* q if there exists a modal refinement relation $\mathscr{R}$ such that $(p,q) \in \mathscr{R}$. Moreover, we denote the kernel of $\mathrel{\overset{\frown}{\sqsubseteq}}$ by $\mathrel{\overset{\frown}{\sqsubseteq}}\!\!\mathrel{\overset{\frown}{\sqsupseteq}}$.

3.1 Parallel Composition

The definitions of composability, parallel product and error state for Relaxed MIAs are as in Def. 4 for MIAs. However, the pessimistic setting is distinguished from the optimistic one by the following definition of *compatibility*, which is much stricter than the notion of compatibility introduced in Def. 4:

Definition 13 (Compatibility on Relaxed MIA) Given Relaxed MIAs P_1 and P_2, states $p_1 \in P_1$ and $p_2 \in P_2$ are called *incompatible* if an error state is reachable from (p_1,p_2) in $P_1 \otimes P_2$. Here, *reachable* means reachable via any kind of may-transition. We write $p_1 \otimes p_2$ for (p_1,p_2) if p_1 and p_2 are compatible.

Note that Lemma 1 is still valid in the pessimistic setting. We now obtain the analogue of Thm. 1:

Theorem 4 (Compositionality of Parallel Composition) *Let P_1, P_2, Q be Relaxed MIAs with $p_1 \in P_1$, $p_2 \in P_2$, $q \in Q$ and $p_1 \sqsubseteq_{\frown} q$. Assume that Q and P_2 are composable; then:*

(a) *P_1 and P_2 are composable.*

(b) *If q and p_2 are compatible, then so are p_1 and p_2 and $p_1 \otimes p_2 \sqsubseteq_{\frown} q \otimes p_2$.*

In contrast to the optimistic setting, the matching of input may-transitions in the refinement preorder does not preclude compositionality. This is because for $p_1 \sqsubseteq_{\frown} q$, there exist much fewer p_2 such that q and p_2 are compatible. In other words, for establishing the precongruence property for parallel composition $\otimes$, there are much fewer results $p_1 \otimes p_2 \sqsubseteq_{\frown} q \otimes p_2$ to prove.

3.2 Conjunction & Disjunction

The definition of conjunction $\wedge$ on Relaxed MIA gets by with five instead of eleven rules. This is because it allows one to merge the corresponding rules for outputs and inputs, due to the use of weak input must-transitions in Def. 12:

Definition 14 (Conjunctive Product on Relaxed MIA) Let $(P,I,O,\longrightarrow_P,\dashrightarrow_P)$ and $(Q,I,O,\longrightarrow_Q,\dashrightarrow_Q)$ be Relaxed MIAs with common alphabets. The conjunctive product $P\&Q =_{\mathrm{df}} (P\times Q,I,O,\longrightarrow,\dashrightarrow)$ is defined by the following operational transition rules:

(Must1)	$(p,q) \xrightarrow{\alpha} \{(p',q') \mid p' \in P', q \stackrel{\hat{\alpha}}{=\!=\!\Rightarrow}_Q q'\}$	if	$p \xrightarrow{\alpha}_P P'$ and $q \stackrel{\hat{\alpha}}{=\!=\!\Rightarrow}_Q$
(Must2)	$(p,q) \xrightarrow{\alpha} \{(p',q') \mid p \stackrel{\hat{\alpha}}{=\!=\!\Rightarrow}_P p', q' \in Q'\}$	if	$p \stackrel{\hat{\alpha}}{=\!=\!\Rightarrow}_P$ and $q \xrightarrow{\alpha}_Q Q'$
(May1)	$(p,q) \stackrel{\tau}{\dashrightarrow} (p',q)$	if	$p \stackrel{\tau}{=\!=\!\Rightarrow}_P p'$
(May2)	$(p,q) \stackrel{\tau}{\dashrightarrow} (p,q')$	if	$q \stackrel{\tau}{=\!=\!\Rightarrow}_Q q'$
(May3)	$(p,q) \stackrel{\alpha}{\dashrightarrow} (p',q')$	if	$p \stackrel{\alpha}{=\!=\!\Rightarrow}_P p'$ and $q \stackrel{\alpha}{=\!=\!\Rightarrow}_Q q'$

Conjunction on Relaxed MIA – including the set F of inconsistent states – is now defined identically to these notions on MIA (Def. 7), but replacing $o \in O$ with $a \in A$; the same applies to the notion of witness (Def. 8). In analogy to Lemma 2, we obtain the following concrete witness lemma for our pessimistic setting:

Lemma 3 (Concrete Witness for Relaxed MIAs) *Let P, Q and R be Relaxed MIAs.*

(i) *For any witness W of $P\&Q$, we have $F \cap W = \emptyset$.*

(ii) *The set $\{(p,q) \in P\times Q \mid \exists r \in R.\, r \sqsubseteq_{\frown} p$ and $r \sqsubseteq_{\frown} q\}$ is a witness of $P\&Q$.*

On the basis of this lemma we can now establish the desired greatest lower bound result for $\wedge$, which implies the compositionality of $\sqsubseteq_{\frown}$ wrt. $\wedge$:

Theorem 5 ($\wedge$ is And) *Let P and Q be Relaxed MIAs with common alphabets. Then, (i) ($\exists R$ and $r \in R$. $r \sqsubseteq_{\frown} p$ and $r \sqsubseteq_{\frown} q$) iff $p \wedge q$ defined. Further, in case $p \wedge q$ defined and for any R and $r \in R$: (ii) $r \sqsubseteq_{\frown} p$ and $r \sqsubseteq_{\frown} q$ iff $r \sqsubseteq_{\frown} p \wedge q$.*

Corollary 3 *Modal-refinement is compositional wrt. conjunction.*

We now turn our attention to disjunction $\vee$ on Relaxed MIA, which is defined as in Def. 9 for MIA and for which we obtain, in analogy to Thm. 3 and Cor. 3:

Theorem 6 ($\vee$ is Or) *Let P, Q and R be Relaxed MIAs with common alphabets, disjoint state sets and states p, q and r, resp. Then, $p \vee q \sqsubseteq\!\!\!\!\frown\; r$ iff $p \sqsubseteq\!\!\!\!\frown\; r$ and $q \sqsubseteq\!\!\!\!\frown\; r$.*

Corollary 4 *Modal-refinement is compositional wrt. disjunction.*

3.3 Alphabet Extension

As motivated in Sec. 2, we introduce alphabet extension as an operation on Relaxed MIA:

Definition 15 (Alphabet Extension) Given a Relaxed MIA $(P, I, O, \longrightarrow, \dashrightarrow)$ and disjoint action sets I' and O' satisfying $I' \cap A = \emptyset = O' \cap A$, where $A =_{\text{df}} I \cup O$. The *alphabet extension of P by I' and O'* is given by $[P]_{I',O'} =_{\text{df}} (P, I \cup I', O \cup O', \longrightarrow, \dashrightarrow')$ for $\dashrightarrow' =_{\text{df}} \dashrightarrow \cup \{(p,a,p) \mid p \in P, a \in I' \cup O'\}$. We often write $[p]_{I',O'}$ – or conveniently $[p]$ in case I', O' are understood from the context – for p as state of $[P]_{I',O'}$.

For Relaxed MIAs P, Q with $p \in P$, $q \in Q$, $I_P \supseteq I_Q$ and $O_P \supseteq O_Q$, we define $p \sqsubseteq\!\!\!\!\frown'\; q$ if $p \sqsubseteq\!\!\!\!\frown\; [q]_{I_P \setminus I_Q, O_P \setminus O_Q}$. Since $\sqsubseteq\!\!\!\!\frown'$ extends $\sqsubseteq\!\!\!\!\frown$ to Relaxed MIAs with different alphabets, we write $\sqsubseteq\!\!\!\!\frown$ for $\sqsubseteq\!\!\!\!\frown'$. We also abbreviate $[q]_{I_P \setminus I_Q, O_P \setminus O_Q}$ by $[q]_P$.

Our compositionality result regarding parallel composition of Thm. 4 immediately carries over to the alphabet extension situation, if we require that alphabet extension does not yield new communications:

Theorem 7 (Compositionality of Parallel Composition) *Let P_1, P_2, Q be Relaxed MIAs as well as $p_1 \in P_1$, $p_2 \in P_2$, $q \in Q$ such that, for $I' =_{df} I_1 \setminus I_Q$ and $O' =_{df} O_1 \setminus O_Q$, we have $(I' \cup O') \cap A_2 = \emptyset$. Assume further that Q and P_2 are composable and $p_1 \sqsubseteq\!\!\!\!\frown\; q$. Then:*

(a) *P_1 and P_2 are composable.*

(b) *If q and p_2 are compatible, then so are p_1 and p_2 and $p_1 \otimes p_2 \sqsubseteq\!\!\!\!\frown\; q \otimes p_2$.*

The conjunction operator in the presence of alphabet extension can now be lifted from Sec. 3.2 in a straightforward manner:

Definition 16 (Conjunction Operator) Let P, Q be Relaxed MIAs, $p \in P$ and $q \in Q$ such that $I_P \cap O_Q = \emptyset = I_Q \cap O_P$. Then, $p \wedge' q =_{\text{df}} [p]_Q \wedge [q]_P$. Again, we simply write $p \wedge q$ for $p \wedge' q$.

To be able to lift our main result, Thm. 5, we only need to establish that the alphabet extension operation is a homomorphism for conjunction:

Lemma 4 *Let P with $p \in P$ and Q with $q \in Q$ be Relaxed MIAs that have the same alphabets. Consider the alphabet extensions by some I' and O'. Then:*

(a) *p and q are consistent iff $[p]$ and $[q]$ are.*

(b) *Given consistency, $[p \wedge q] \sqsubset\!\sqsupset [p] \wedge [q]$.*

Theorem 8 ($\wedge$ is And) *Let P with $p \in P$, Q with $q \in Q$, and R with $r \in R$ be Relaxed MIAs such that $I_P \cap O_Q = \emptyset = I_Q \cap O_P$, $I_R \supseteq I_P \cup I_Q$ and $O_R \supseteq O_P \cup O_Q$. Then, (i) there exists such an R and $r \in R$ with $r \sqsubset p$ and $r \sqsubset q$ iff $p \wedge q$ is defined. Further, in case $p \wedge q$ is defined: (ii) $r \sqsubset p$ and $r \sqsubset q$ iff $r \sqsubset p \wedge q$.*

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

Definition 17 (Disjunction Operator) Let P, Q be Relaxed MIAs with disjoint state sets, $p \in P$ and $q \in Q$ such that $I_P \cap O_Q = \emptyset = I_Q \cap O_P$. Then, $p \vee' q =_{\mathrm{df}} [p]_Q \vee [q]_P$. Once again, we simply write $p \vee q$ for $p \vee' q$.

Lemma 5 *Let P with $p \in P$ and R with $r \in R$ be Relaxed MIAs having the same alphabets, as well as I' and O' be suitable action sets for extending them. Then, $p \sqsubset r$ iff $[p] \sqsubset [r]$.*

Theorem 9 ($\vee$ is Or) *Let P with $p \in P$, Q with $q \in Q$, and R with $r \in R$ be Relaxed MIAs with disjoint state sets 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, $p \vee q \sqsubset r$ iff $p \sqsubset r$ and $q \sqsubset r$.*

3.4 Example

We briefly illustrate the utility of our interface theory by a small example that models parts of a communication protocol (see Fig. 4) and is inspired by an example in [RBB+11]. The protocol's abstract specification is given by MIA `Spec`. It receives a message from its environment (action `get?`), delivers it (`put!`) and signals to its environment its willingness to handle the next message (`nxt!`). The two τ-may-transitions making up the τ-loop model that the message's transmission may fail and that this failure may possibly be repaired.

The design of our communication protocol contains a generic component `Sender`, which receives a message for delivery (`get?`). It sends this message (`msg!`) to the `Medium` and waits for an according acknowledgment (`ack?`). In case a negative acknowledgment arrives (`nack?`), the message is re-sent. `Sender` is specialized by conjoining it with component `Resetter`, which can suggest a reset (`rst!`) after a negative acknowledgment; it is defined such that an implementation may decide, e.g., to initiate a reset after exactly n negative acknowledgments, showing the utility of may-inputs for specification and the model-refinement preorder.

(Relaxed) MIA `Sender`$\wedge$`Resetter` is the result of formally applying our conjunction operator to (Relaxed) MIA `Sender` and Relaxed MIA `Resetter`. Note that applying conjunction implicitly extends the alphabet of `Resetter` by `get?`, `ack?`, `msg!` and `nxt!` (cf. Def. 16). In addition, no inconsistency arises in our example. However, if one would refine `Sender` and `Resetter` by removing the `rst!`-loop at state D and making the reset transition a must-transition instead of a may-transition, then state `Db` (or, more precisely, `D`$\wedge$`b`) would be

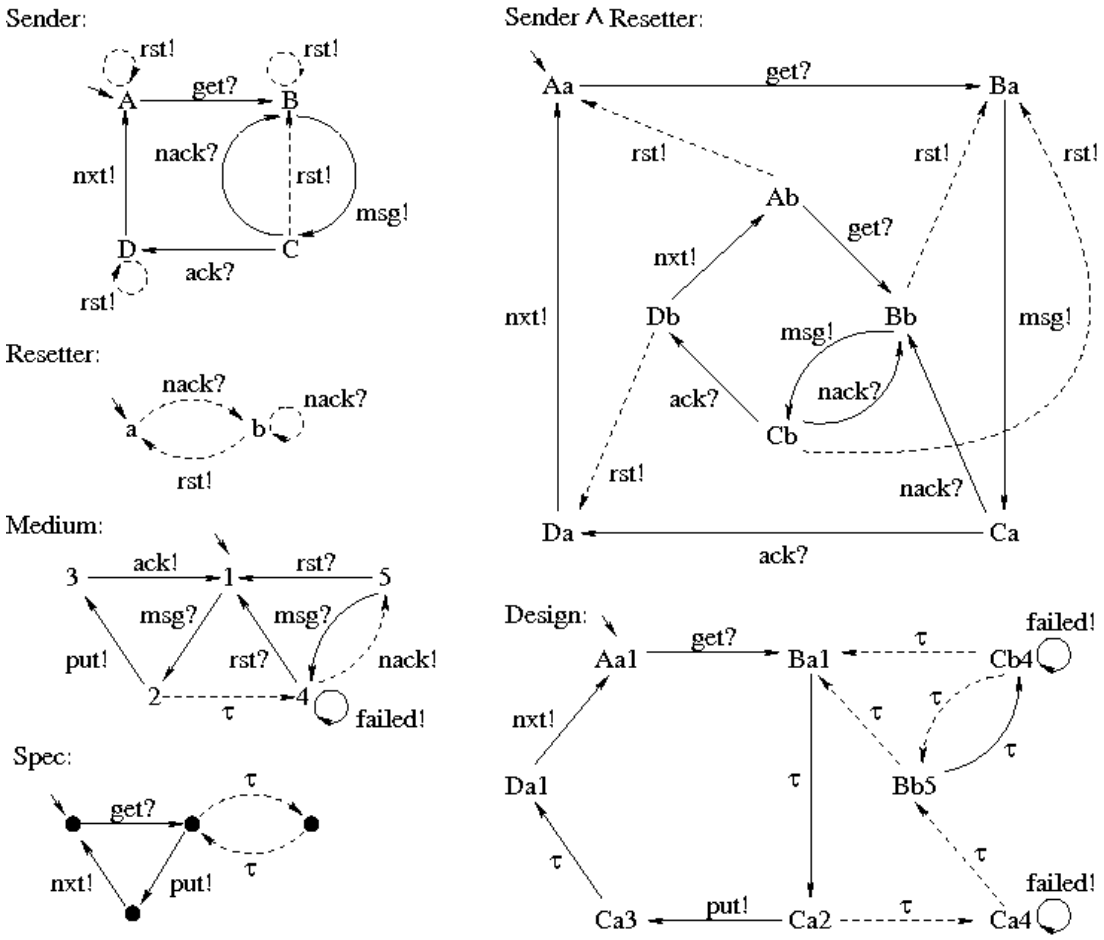

Figure 4: Example: `Design` = (`Sender` ∧ `Resetter`)|`Medium` and `Spec`.

inconsistent; removing this state as is required by the definition of ∧ would then yield state `Ab` unreachable.

(Relaxed) MIA `Medium` specifies a communication medium with potential failure, which receives a message (`msg?`) and may either deliver it to the environment (`put!`) or – via the τ-may-transition – may lose it. In the former case, `Medium` returns to its initial state by sending an acknowledgment (`ack!`); in the latter case, it signals failure (`failed!`) and may return a negative acknowledgment (`nack!`). The parallel composition `Design` $=_{\mathrm{df}}$ (`Sender` ∧ `Resetter`)|`Medium` is also shown in Fig. 4. Using our refinement preorder, it is now easy to check that `Design` modal-refines `Spec` when the latter's alphabet is extended by `failed!`, i.e., `Design` $\sqsubseteq$ $[$`Spec`$]_{\emptyset,\{\texttt{failed}\}}$. Note that this relies on our use of *weak* must- and may-transitions in the definition of $\sqsubseteq$, i.e., on the ability to abstract from internal τ-transitions.

As a practical case for the problem that we discussed using Fig. 3, observe the following: if `Resetter` would permit everything after the unknown input `get?`, output `rst!` would then always be allowed – also without any occurrence of `nack?`.

4 Conclusions & Future Work

Interface theories are an important tool for reasoning about component-based systems [BDH+12, BMSH10, BCHS07, dH05, LNW07, LV13, RBB+11]. This paper advanced the state-of-the-art of both the optimistic and pessimistic schools on interface theories. Regarding the optimistic school, we repaired a shortcoming of the refinement preorder introduced in [LNW07], which ignored internal must-transitions, thereby leading to unintuitive refinements. Regarding the pessimistic school, we showed how its approach may be extended by conjunction and disjunction operators; conjunction is a key operator in any component-based setting, which enables engineers to express that some component is required to satisfy several interfaces.

In future work we wish to investigate whether there are suitable interface theories in-between the optimistic and pessimistic approaches. This might fix their current limitations, namely allowing may-inputs as in the pessimistic approach while maintaining the truly *open systems* view of the optimistic approach.

Bibliography

[Bau12] S. Bauer. *Modal Specification Theories for Component-based Design*. PhD thesis, Faculty of Mathematics, Informatics and Statistics, LMU Munich, Germany, 2012.

[BCHS07] D. Beyer, A. Chakrabarti, T. Henzinger, S. Seshia. An Application of Web-Service Interfaces. In *ICWS*. Pp. 831–838. IEEE, 2007.

[BDH+12] S. Bauer, A. David, R. Hennicker, K. Larsen, A. Legay, U. Nyman, A. Wasowski. Moving from Specifications to Contracts in Component-Based Design. In *FASE*. LNCS 7212, pp. 43–58. Springer, 2012.

[BMSH10] S. Bauer, P. Mayer, A. Schroeder, R. Hennicker. On Weak Modal Compatibility, Refinement, and the MIO Workbench. In *TACAS*. LNCS 6015, pp. 175–189. Springer, 2010.

[dH05] L. de Alfaro, T. Henzinger. Interface-based Design. In *Engineering Theories of Software-Intensive Systems*. NATO Science Series 195. Springer, 2005.

[Lar90] K. Larsen. Modal Specifications. In *Automatic Verification Methods for Finite State Systems*. LNCS 407, pp. 232–246. Springer, 1990.

[LNW07] K. Larsen, U. Nyman, A. Wasowski. Modal I/O Automata for Interface and Product Line Theories. In *ESOP*. LNCS 4421, pp. 64–79. Springer, 2007.

[LV13] G. Lüttgen, W. Vogler. Modal Interface Automata. *Logical Methods in Computer Science* 9(3:4), 2013.

[RBB+11] J. Raclet, E. Badouel, A. Benveniste, B. Caillaud, A. Legay, R. Passerone. A Modal Interface Theory for Component-based Design. *Fund. Inform.* 107:1–32, 2011.