Logical vs. Behavioural Specifications

Nikola Beneš, Uli Fahrenberg, Jan Křetínský, Axel Legay, Louis-Marie Traonouez

Logical vs. Behavioural Specifications

Nikola Beneš[a], Uli Fahrenberg[b,*], Jan Křetínský[c], Axel Legay[b], Louis-Marie Traonouez[b]

[a] *Masaryk University Brno*
[b] *Irisa / Inria Rennes*
[c] *IST Austria*

Abstract

There are two fundamentally different approaches to specifying and verifying properties of systems. The *logical* approach makes use of specifications given as formulae of temporal or modal logics and relies on efficient model checking algorithms; the *behavioural* approach exploits various equivalence or refinement checking methods, provided the specifications are given in the same formalism as implementations.

In this paper we provide translations between the logical formalism of ν-calculus and the behavioural formalism of disjunctive modal transition systems. The translations preserve structural properties of the specification and allow us to perform logical operations on the behavioural specifications as well as behavioural compositions on logical formulae. The unification of both approaches provides additional methods for component-based stepwise design.

Keywords: component-based design, refinement, logic, modal transition system, specification

1. Introduction

There are two fundamentally different approaches to specifying and verifying properties of systems. Firstly, the *logical* approach makes use of specifications given as formulae of temporal or modal logics and relies on efficient model checking algorithms. Secondly, the *behavioural* approach exploits various equivalence or refinement checking methods, provided the specifications are given in the same formalism as implementations.

In this paper, we discuss different specification formalisms and their relationship. As an example, let us consider labelled transition systems and the property that *"at all time points after executing* req*, no* idle *nor further requests but only* work *is allowed until* grant *is executed"*. The property can be written in *e.g.* CTL [17] as

$$\mathrm{AG}(\mathsf{req} \Rightarrow \mathrm{AX}(\mathsf{work}\ \mathrm{AW}\ \mathsf{grant}))$$

[☆] This article is based on the conference papers [5, 20] which have been presented at the 24th International Conference on Concurrency Theory in Buenos Aires, Argentina, and at the 11th International Colloquium on Theoretical Aspects of Computing in Bucharest, Romania.

[*] Corresponding author. Irisa / Inria Rennes, Campus Beaulieu, 35042 Rennes Cedex, France. *Phone number:* +33.6.99.84.22.75. *Email address:* ulrich.fahrenberg@inria.fr

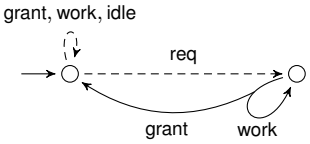

Figure 1: DMTS corresponding to the CTL property AG(req ⇒ AX(work AW grant))

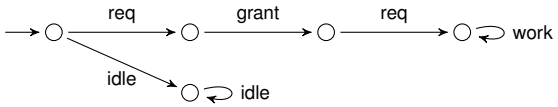

Figure 2: An implementation of the specification in Fig. 1

or as a recursive system of equations in Hennessy-Milner logic [35] as

$$X = [\mathsf{grant}, \mathsf{idle}, \mathsf{work}]X \wedge [\mathsf{req}]Y$$
$$Y = (\langle \mathsf{work} \rangle Y \vee \langle \mathsf{grant} \rangle X) \wedge [\mathsf{idle}, \mathsf{req}]\mathbf{ff}$$

where the solution is given by the greatest fixed point.

As formulae of modal logics can be difficult to read, some people prefer automata-based behavioural specifications to logical ones. One such behavioural specification formalism is the one of disjunctive modal transition systems (DMTS) [37]. Figure 1 displays a specification of our example property as a DMTS. Here the dashed arrows indicate that the transitions *may or may not* be present, while branching of the solid arrow indicates that at least one of the branches *must* be present. An example of a labelled transition system that *satisfies* our logical specifications and *implements* the behavioural one is given in Fig. 2.

The alternative between logical and behavioural specifications is not only a question of preference. Logical specification formalisms put a powerful logical language at the disposal of the user, and the logical approach to model checking [17, 43] has seen a lot of success and tool implementations. Automata-based specifications [13, 33], on the other hand, have a focus on *compositional* and *incremental* design. For a model consisting of several components, compositionality allows us to (1) infer properties of a system from the specifications of its components, and (2) decompose the problem of correctness for a system into verification problems for its components, which helps in overcoming the state space explosion problem. In this respect, logical specifications are somewhat lacking. Further, given a global property of a model and a component of the model that is already known to satisfy a local property, one would be able to *decompose* automatically, from the global property and the local property, a new property which the rest of the model must satisfy. We refer to [34] for a good account of composition and decomposition and other features which one would wish specifications to have. It is thus desirable to be able to translate specifications from the logical realm into behavioural formalisms, and *vice versa* from behavioural formalisms to logic-based specifications.

$$X = \langle a \rangle \mathbf{tt} \wedge [a]X \wedge [b]X$$

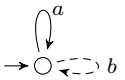

Figure 3: ν-calculus formula and DMTS for the invariance property "*there is always an 'a' transition available*", over the alphabet $\Sigma = \{a, b\}$

$$X = \langle b \rangle \mathbf{tt} \vee \big(\langle a \rangle \mathbf{tt} \wedge [a]X \wedge [b]X \wedge [c]X \big)$$

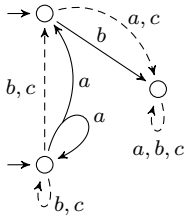

Figure 4: ν-calculus formula and DMTS for the ("weak until") property "*there is always an 'a' transition available, until a 'b' transition becomes enabled*", over the alphabet $\Sigma = \{a, b, c\}$

Our Contribution

Firstly, we show that Hennessy-Milner logic with greatest fixed points (the modal ν-calculus) and DMTS (with several initial states) are equally expressive, and we provide translations forth and back. Moreover, the translations preserve *structural* properties of the formulae/systems and enable us to freely combine the two formalisms. For doing this, we introduce an auxiliary intermediate formalism NAA (a nondeterministic extension of acceptance automata [27, 45]) and a related hybrid modal logic [9, 42], so that both turn out to be equivalent in expressiveness to ν-calculus and DMTS. The established connection also allows for a graphical representation of ν-calculus as DMTS. This extends the graphical representability of Hennessy-Milner logic (without fixed points) as modal transition systems [11, 33].

Example 1. Figures 3 and 4 show examples of important basic properties expressed both as formulae in the modal ν-calculus and as DMTS. We refer to Section 2 for the formal definitions. Note that the DMTS in Fig. 4 has *two* initial states; we will show in Section 3.5 that allowing multiple initial states is essential for the correspondence between DMTS and the modal ν-calculus.

Secondly, we show that there are natural operations of conjunction and disjunction for DMTS that mimic the ones of ν-calculus. As we work with multiple initial states, disjunction is readily defined, and conjunction extends the one for DMTS with single initial state [4].

Thirdly, we introduce compositionality into the formalisms. For simplicity we assume CSP-style synchronisation of labels, but the construction can easily be generalised to other types of label synchronisation. We define a composition operator for specifications which soundly captures parallel composition of implementations, and we provide a solution to the open problem of the general quotient.

The intuition of quotient, or decomposition, is as follows. Given a specification S of a final system to be constructed and T either an already implemented component or a specification of a service to be used, the task is to construct the most general specification of the rest of the system to be implemented, in

such a way that when composed with any implementation of T, it conforms with the specification S. This specification is exactly the quotient S/T. We extend the quotient constructions for deterministic modal transition systems (MTS) and acceptance automata [45] to define the quotient for the full class of (possibly nondeterministic) DMTS. We also provide a more efficient procedure for (possibly nondeterministic) MTS. These constructions are the technically most demanding parts of the paper.

With the operations of composition and quotient, all four discussed formalisms form *commutative residuated lattices* [26, 51] (up to equivalence). This makes a rich algebraic theory available for compositional reasoning about specifications. Moreover, they form *complete specification theories* in the sense of [2]. Hence they support full compositionality and decomposition in the sense of [34]. Using our translations, we can transport these notions to the modal ν-calculus, thus also turning the modal ν-calculus into a complete specification theory.

Outline of the Paper

Section 2 introduces all the formalisms of DMTS, ν-calculus and the auxiliary formalisms of NAA and a new hybrid modal logic, which can serve as compact representation for NAA and is of interest in itself. In Section 3 the equivalence of the systems is discussed. In Section 4, we use our translations to turn the formalisms into complete specification theories. Section 5 discusses related work and Section 6 concludes.

2. Specification Formalisms

In this section we introduce the four specification formalisms with which this paper is concerned. For the rest of the paper, we fix a finite alphabet Σ. In each of the formalisms, the semantics of a specification is a set of implementations, in our case always a set of *(finite) labelled transition systems* (LTS) over Σ, *i.e.* structures $\mathcal{I} = (S, s^0, \longrightarrow)$ consisting of a finite set S of *states*, an initial state $s^0 \in S$, and a *transition relation* $\longrightarrow \subseteq S \times \Sigma \times S$.

2.1. Disjunctive Modal Transition Systems

Definition 1. A *disjunctive modal transition system* (DMTS) is a structure $\mathcal{D} = (S, S^0, \dashrightarrow, \longrightarrow)$ consisting of finite sets $S \supseteq S^0$ of states and initial states, a *may*-transition relation $\dashrightarrow \subseteq S \times \Sigma \times S$, and a *disjunctive must*-transition relation $\longrightarrow \subseteq S \times 2^{\Sigma \times S}$. It is assumed that for all $(s, N) \in \longrightarrow$ and all $(a, t) \in N$, $(s, a, t) \in \dashrightarrow$.

As customary, we write $s \overset{a}{\dashrightarrow} t$ instead of $(s, a, t) \in \dashrightarrow$, $s \longrightarrow N$ instead of $(s, N) \in \longrightarrow$, $s \overset{a}{\dashrightarrow}$ if there exists t for which $s \overset{a}{\dashrightarrow} t$, and $s \overset{a}{\not\dashrightarrow}$ if there does not.

The intuition is that may-transitions $s \overset{a}{\dashrightarrow} t$ specify which transitions are permitted in an implementation, whereas a must-transition $s \longrightarrow N$ stipulates a disjunctive requirement: at least one of the choices $(a, t) \in N$ has to be implemented. A DMTS $(S, S^0, \dashrightarrow, \longrightarrow)$ is an *implementation* if $S^0 = \{s^0\}$ is a singleton and $\longrightarrow = \{(s, \{(a, t)\}) \mid s \overset{a}{\dashrightarrow} t\}$.

DMTS were introduced in [37] in the context of equation solving. They are a natural closure of *modal transition systems* (MTS) [33]. We say that a DMTS $(S, S^0, \dashrightarrow, \longrightarrow)$ is a MTS if $S^0 = \{s^0\}$ is a singleton and for all $s \longrightarrow N$ it holds

that $N = \{(a,t)\}$ is also a singleton. When speaking about MTS, we usually write $s \xrightarrow{a} t$ instead of $s \longrightarrow \{(a,t)\}$.

An LTS $(S, s^0, \longrightarrow)$ can be translated to a DMTS implementation $(S, S^0, \dashrightarrow, \longrightarrow')$ by setting $S^0 = \{s^0\}$, $\dashrightarrow = \longrightarrow$ and $\longrightarrow' = \{(s, \{(a,t)\}) \mid s \xrightarrow{a} t\}$. This defines an embedding of LTS into DMTS whose image are precisely the DMTS implementations.

Definition 2. Let $\mathcal{D}_1 = (S_1, S_1^0, \dashrightarrow_1, \longrightarrow_1)$ and $\mathcal{D}_2 = (S_2, S_2^0, \dashrightarrow_2, \longrightarrow_2)$ be DMTS. A relation $R \subseteq S_1 \times S_2$ is a *modal refinement* if it holds for all $(s_1, s_2) \in R$ that

- for all $s_1 \overset{a}{\dashrightarrow} t_1$ there is $t_2 \in S_2$ with $s_2 \overset{a}{\dashrightarrow} t_2$ and $(t_1, t_2) \in R$, and
- for all $s_2 \longrightarrow N_2$ there is $s_1 \longrightarrow N_1$ such that for each $(a, t_1) \in N_1$ there is $(a, t_2) \in N_2$ with $(t_1, t_2) \in R$.

We say that $\mathcal{D}_1$ *modally refines* $\mathcal{D}_2$, denoted $\mathcal{D}_1 \leq_{\mathsf{m}} \mathcal{D}_2$, whenever there exists a modal refinement R such that for all $s_1^0 \in S_1^0$, there exists $s_2^0 \in S_2^0$ for which $(s_1^0, s_2^0) \in R$.

We write $\mathcal{D}_1 \equiv_{\mathsf{m}} \mathcal{D}_2$ if $\mathcal{D}_1 \leq_{\mathsf{m}} \mathcal{D}_2$ and $\mathcal{D}_2 \leq_{\mathsf{m}} \mathcal{D}_1$. For states $s_1 \in S_1$, $s_2 \in S_2$, we write $s_1 \leq_{\mathsf{m}} s_2$ if the DMTS $(S_1, \{s_1\}, \dashrightarrow_1, \longrightarrow_1) \leq_{\mathsf{m}} (S_2, \{s_2\}, \dashrightarrow_2, \longrightarrow_2)$. Sometimes we will refer to the last property of a modal refinement relation, $\forall s_1^0 \in S_1^0 : \exists s_2^0 \in S_2^0 : (s_1^0, s_2^0) \in R$, as being *initialised*. Note that modal refinement is reflexive and transitive, *i.e.* a preorder on DMTS.

The *set of implementations* of a DMTS $\mathcal{D}$ is $[\![\mathcal{D}]\!] = \{\mathcal{I} \leq_{\mathsf{m}} \mathcal{D} \mid \mathcal{I} \text{ implementation}\}$. This is, thus, the set of all LTS which satisfy the specification given by the DMTS $\mathcal{D}$. We say that $\mathcal{D}_1$ *thoroughly refines* $\mathcal{D}_2$, and write $\mathcal{D}_1 \leq_{\mathsf{t}} \mathcal{D}_2$, if $[\![\mathcal{D}_1]\!] \subseteq [\![\mathcal{D}_2]\!]$. We write $\mathcal{D}_1 \equiv_{\mathsf{t}} \mathcal{D}_2$ if $\mathcal{D}_1 \leq_{\mathsf{t}} \mathcal{D}_2$ and $\mathcal{D}_2 \leq_{\mathsf{t}} \mathcal{D}_1$. For states $s_1 \in S_1$, $s_2 \in S_2$, we write $[\![s_1]\!] = [\![(S_1, \{s_1\}, \dashrightarrow_1, \longrightarrow_1)]\!]$ and $s_1 \leq_{\mathsf{t}} s_2$ if $[\![s_1]\!] \subseteq [\![s_2]\!]$.

The below proposition, which follows directly from transitivity of modal refinement, shows that modal refinement is *sound* with respect to thorough refinement; in the context of specification theories, this is what one would expect, and we only include it for completeness of presentation. It can be shown that modal refinement is also *complete* for *deterministic* DMTS [4], but we will not need this here.

Proposition 3. *For all DMTS $\mathcal{D}_1$, $\mathcal{D}_2$, $\mathcal{D}_1 \leq_{\mathsf{m}} \mathcal{D}_2$ implies $\mathcal{D}_1 \leq_{\mathsf{t}} \mathcal{D}_2$.* □

2.2. The Modal ν-Calculus

We recall the syntax and semantics of the modal ν-calculus, the fragment of the modal μ-calculus [30, 49] with only maximal fixed points. Instead of an explicit maximal fixed point operator, we use the representation by equation systems in Hennessy-Milner logic developed in [35].

For a finite set X of variables, let $\mathcal{H}(X)$ be the set of *Hennessy-Milner formulae*, generated by the abstract syntax $\mathcal{H}(X) \ni \phi ::= \mathbf{tt} \mid \mathbf{ff} \mid x \mid \langle a \rangle \phi \mid [a]\phi \mid \phi \wedge \phi \mid \phi \vee \phi$, for $a \in \Sigma$ and $x \in X$.

A *declaration* is a mapping $\Delta : X \to \mathcal{H}(X)$; we recall the maximal fixed point semantics of declarations from [35]. Let $(S, s^0, \longrightarrow)$ be an LTS, then an *assignment* is a mapping $\sigma : X \to 2^S$. The set of assignments forms

a complete lattice with order $\sigma_1 \sqsubseteq \sigma_2$ iff $\sigma_1(x) \subseteq \sigma_2(x)$ for all $x \in X$ and lowest upper bound $\big(\bigsqcup_{i\in I} \sigma_i\big)(x) = \bigcup_{i\in I} \sigma_i(x)$.

The semantics of a formula is a subset of S, given relative to an assignment σ, defined as follows: $(\!|\mathbf{tt}|\!)\sigma = S$, $(\!|\mathbf{ff}|\!)\sigma = \emptyset$, $(\!|x|\!)\sigma = \sigma(x)$, $(\!|\phi \wedge \psi|\!)\sigma = (\!|\phi|\!)\sigma \cap (\!|\psi|\!)\sigma$, $(\!|\phi \vee \psi|\!)\sigma = (\!|\phi|\!)\sigma \cup (\!|\psi|\!)\sigma$, and

$$(\!|\langle a\rangle\phi|\!)\sigma = \{s \in S \mid \exists s \xrightarrow{a} s' : s' \in (\!|\phi|\!)\sigma\},$$
$$(\!|[a]\phi|\!)\sigma = \{s \in S \mid \forall s \xrightarrow{a} s' : s' \in (\!|\phi|\!)\sigma\}.$$

The semantics of a declaration Δ is then the assignment defined by

$$(\!|\Delta|\!) = \bigsqcup\{\sigma : X \to 2^S \mid \forall x \in X : \sigma(x) \subseteq (\!|\Delta(x)|\!)\sigma\};$$

the maximal (pre)fixed point of Δ.

A *ν-calculus expression* is a structure $\mathcal{N} = (X, X^0, \Delta)$, with $X^0 \subseteq X$ sets of variables and $\Delta : X \to \mathcal{H}(X)$ a declaration. We say that an LTS $\mathcal{I} = (S, s^0, \longrightarrow)$ *implements* (or models) the expression, and write $\mathcal{I} \models \mathcal{N}$, if there is $x^0 \in X^0$ such that $s^0 \in (\!|\Delta|\!)(x^0)$. We write $[\![\mathcal{N}]\!]$ for the set of implementations (models) of a ν-calculus expression $\mathcal{N}$. As for DMTS, we write $[\![x]\!] = [\![(X, \{x\}, \Delta)]\!]$ for $x \in X$, and thorough refinement of expressions and states is defined accordingly.

We are now going to introduce a *normal form* for ν-calculus expressions. The purpose of this normal form is twofold. One is to allow us to define modal refinement for ν-calculus, an analogue to the DMTS modal refinement that can be seen as a sound approximation of the logical implication, *cf.* Proposition 3. The second purpose is to facilitate a simple translation between DMTS and ν-calculus expressions, see Section 3.1 below.[1]

Lemma 4. *For any ν-calculus expression $\mathcal{N}_1 = (X_1, X_1^0, \Delta_1)$, there exists another expression $\mathcal{N}_2 = (X_2, X_2^0, \Delta_2)$ with $[\![\mathcal{N}_1]\!] = [\![\mathcal{N}_2]\!]$ and such that for any $x \in X$, $\Delta_2(x)$ is of the form*

$$\Delta_2(x) = \bigwedge_{i\in I}\Big(\bigvee_{j\in J_i}\langle a_{ij}\rangle x_{ij}\Big) \wedge \bigwedge_{a\in\Sigma}[a]\Big(\bigvee_{j\in J_a} y_{a,j}\Big) \tag{1}$$

for finite (possibly empty) index sets I, J_i, J_a, for $i \in I$ and $a \in \Sigma$, and all $x_{ij}, y_{a,j} \in X_2$. Additionally, for all $i \in I$ and $j \in J_i$, there exists $j' \in J_{a_{ij}}$ for which $x_{ij} \leq_{\mathrm{t}} y_{a_{ij},j'}$.

Proof. It is shown in [11] that any Hennessy-Milner formula is equivalent to one in so-called *strong normal form*, *i.e.* of the form $\bigvee_{i\in I}(\bigwedge_{j\in J_i}\langle a_{ij}\rangle\phi_{ij} \wedge \bigwedge_{a\in\Sigma}[a]\psi_{i,a})$ for HML formulas ϕ_{ij}, $\psi_{i,a}$ which are also in strong normal form. Now we can replace the ϕ_{ij}, $\psi_{i,a}$ by (new) variables x_{ij}, $y_{i,a}$ and add declarations $\Delta_2(x_{ij}) = \phi_{ij}$, $\Delta_2(y_{i,a}) = \psi_{i,a}$ to arrive at an expression in which all formulae are of the form $\Delta_2(x) = \bigvee_{i\in I}(\bigwedge_{j\in J_i}\langle a_{ij}\rangle x_{ij} \wedge \bigwedge_{a\in\Sigma}[a]x_{i,a})$.

Now for each such formula, replace (recursively) x by new variables $\{\tilde{x}^i \mid i \in I\}$ and set $\Delta_2(\tilde{x}^i) = \bigwedge_{j\in J_i}\langle a_{ij}\rangle\big(\bigvee_k \tilde{x}^k_{ij}\big) \wedge \bigwedge_{a\in\Sigma}[a]\big(\bigvee_k \tilde{x}^k_{i,a}\big)$. Using initial variables $X_2^0 = \{\tilde{x}^i \mid x \in X_1^0\}$, the so-constructed ν-calculus expression is equivalent to the original one. □

[1] Note that this normal form is different from the one introduced in [5]. The original normal form did not allow for modal refinement and was less apt for the ν-calculus to DMTS translation.

As this is a type of *conjunctive normal form*, it is clear that translating a ν-calculus expression into normal form may incur an exponential blow-up.

We introduce some notation for ν-calculus expressions in normal form which will make our life easier later. Let $\mathcal{N} = (X, X^0, \Delta)$ be such an expression and $x \in X$, with $\Delta(x) = \bigwedge_{i\in I} \big(\bigvee_{j\in J_i} \langle a_{ij}\rangle x_{ij}\big) \wedge \bigwedge_{a\in\Sigma} [a]\big(\bigvee_{j\in J_a} y_{a,j}\big)$ as in the lemma. Define $\Diamond(x) = \{\{(a_{ij}, x_{ij}) \mid j \in J_i\} \mid i \in I\}$ and, for each $a \in \Sigma$, $\Box^a(x) = \{y_{a,j} \mid j \in J_a\}$. Note that now,

$$\Delta(x) = \bigwedge_{N\in\Diamond(x)} \Big(\bigvee_{(a,y)\in N} \langle a\rangle y\Big) \wedge \bigwedge_{a\in\Sigma} [a]\Big(\bigvee_{y\in\Box^a(x)} y\Big).$$

Definition 5. Let $\mathcal{N}_1 = (X_1, X_1^0, \Delta_1)$, $\mathcal{N}_2 = (X_2, X_2^0, \Delta_2)$ be ν-calculus expressions in normal form and $R \subseteq X_1 \times X_2$. The relation R is a *modal refinement* if it holds for all $(x_1, x_2) \in R$ that

- for all $a \in \Sigma$ and every $y_1 \in \Box_1^a(x_1)$, there is $y_2 \in \Box_2^a(x_2)$ for which $(y_1, y_2) \in R$, and
- for all $N_2 \in \Diamond_2(x_2)$ there is $N_1 \in \Diamond_1(x_1)$ such that for each $(a, y_1) \in N_1$, there exists $(a, y_2) \in N_2$ with $(y_1, y_2) \in R$.

We say that $\mathcal{N}_1$ *modally refines* $\mathcal{N}_2$, denoted $\mathcal{N}_1 \leq_{\mathsf{m}} \mathcal{N}_2$, whenever there exists a modal refinement R such that for every $x_1^0 \in X_1^0$ there exists $x_2^0 \in X_2^0$ for which $(x_1^0, x_2^0) \in R$.

We say that a ν-calculus expression (X, X^0, Δ) in normal form is an *implementation* if $X^0 = \{x^0\}$ is a singleton, $\Diamond(x) = \{\{(a, y)\} \mid y \in \Box^a(x), a \in \Sigma\}$ and $\Box^a(x) = \emptyset$ for all $a \notin \Sigma$, for all $x \in X$.

We can translate an LTS $(S, s^0, \longrightarrow)$ to a ν-calculus expression (S, S^0, Δ) in normal form by setting $S^0 = \{s^0\}$ and $\Diamond(s) = \{\{(a, t)\} \mid s \xrightarrow{a} t\}$ and $\Box^a(s) = \{t \mid s \xrightarrow{a} t\}$ for all $s \in S$, $a \in \Sigma$. Like for DMTS, this defines an embedding of LTS into the modal ν-calculus whose image are precisely the ν-calculus implementations.

We will show below in Theorem 10 that for any LTS $\mathcal{I}$ and any ν-calculus expression $\mathcal{N}$ in normal form, $\mathcal{I} \models \mathcal{N}$ iff $\mathcal{I} \leq_{\mathsf{m}} \mathcal{N}$, hence the fixed-point semantics of [35] and our refinement semantics agree. As a corollary of this result, we get that modal refinement is a sound approximation to logical implication, *i.e.* that $\mathcal{N}_1 \leq_{\mathsf{m}} \mathcal{N}_2$ implies that for all implementations $\mathcal{I}$, $(\mathcal{I} \models \mathcal{N}_1) \Rightarrow (\mathcal{I} \models \mathcal{N}_2)$.

2.3. Nondeterministic Acceptance Automata

Definition 6. A *nondeterministic acceptance automaton* (NAA) is a structure $\mathcal{A} = (S, S^0, \mathrm{Tran})$, with $S \supseteq S^0$ finite sets of states and initial states and $\mathrm{Tran} : S \to 2^{2^{\Sigma\times S}}$ an assignment of *transition constraints*. We assume that for all $s^0 \in S^0$, $\mathrm{Tran}(s^0) \neq \emptyset$.

Acceptance automata were first introduced in [44] (see also [45], where a slightly different language-based approach is taken), based on the notion of acceptance trees in [27]; however, these are *deterministic*. We extend the formalism into nondeterministic setting here. The following notion of modal refinement was introduced in [6].

Definition 7. Let $\mathcal{A}_1 = (S_1, S_1^0, \mathrm{Tran}_1)$ and $\mathcal{A}_2 = (S_2, S_2^0, \mathrm{Tran}_2)$ be NAA. A relation $R \subseteq S_1 \times S_2$ is a *modal refinement* if it holds for all $(s_1, s_2) \in R$ and all $M_1 \in \mathrm{Tran}_1(s_1)$ that there exists $M_2 \in \mathrm{Tran}_2(s_2)$ such that

$$\begin{aligned} &\forall (a, t_1) \in M_1 : \exists (a, t_2) \in M_2 : (t_1, t_2) \in R\,, \\ &\forall (a, t_2) \in M_2 : \exists (a, t_1) \in M_1 : (t_1, t_2) \in R\,. \end{aligned} \tag{2}$$

We say that $\mathcal{A}_1$ *modally refines* $\mathcal{A}_2$, and write $\mathcal{A}_1 \leq_{\mathsf{m}} \mathcal{A}_2$, whenever there exists a modal refinement R such that for all $s_1^0 \in S_1^0$, there exists $s_2^0 \in S_2^0$ for which $(s_1^0, s_2^0) \in R$.

An NAA is an *implementation* if $S^0 = \{s^0\}$ is a singleton and, for all $s \in S$, $\mathrm{Tran}(s) = \{M\}$ is a singleton. An LTS $(S, s^0, \longrightarrow)$ can be translated to an NAA by setting $S^0 = \{s^0\}$ and $\mathrm{Tran}(s) = \{\{(a,t) \mid s \xrightarrow{a} t\}\}$. This defines an embedding of LTS into NAA whose image are precisely the NAA implementations.

2.4. Hybrid Modal Logic

As our fourth specification formalism, we introduce a hybrid modal logic, closely related to the Boolean modal transition systems of [6] and hybrid in the sense of [9, 42]: it contains nominals, and the semantics of a nominal is given as all sets which contain the nominal.

For a finite set X of nominals, let $\mathcal{L}(X)$ be the set of formulae generated by the abstract syntax $\mathcal{L}(X) \ni \phi \coloneqq \mathbf{tt} \mid \mathbf{ff} \mid \langle a \rangle x \mid \neg\phi \mid \phi \wedge \phi$, for $a \in \Sigma$ and $x \in X$. The semantics of a formula is a set of subsets of $\Sigma \times X$, given as follows: $(\!|\mathbf{tt}|\!) = 2^{\Sigma \times X}$, $(\!|\mathbf{ff}|\!) = \emptyset$, $(\!|\neg\phi|\!) = 2^{\Sigma\times X} \setminus (\!|\phi|\!)$, $(\!|\langle a \rangle x|\!) = \{M \subseteq \Sigma \times X \mid (a,x) \in M\}$, and $(\!|\phi \wedge \psi|\!) = (\!|\phi|\!) \cap (\!|\psi|\!)$. We also define disjunction $\phi_1 \vee \phi_2 = \neg(\phi_1 \wedge \phi_2)$.

An *$\mathcal{L}$-expression* is a structure $\mathcal{E} = (X, X^0, \Phi)$ consisting of finite sets $X^0 \subseteq X$ of variables and a mapping $\Phi : X \to \mathcal{L}(X)$. Such an expression is an *implementation* if $(\!|\Phi(x)|\!) = \{M\}$ is a singleton for each $x \in X$.

We can translate an LTS $(S, s^0, \longrightarrow)$ to an $\mathcal{L}$-expression (S, S^0, Φ) by setting $S^0 = \{s^0\}$ and $\Phi(s) = \bigwedge_{s \xrightarrow{a} t} \langle a \rangle t \wedge \bigwedge_{s \not\xrightarrow{b} u} \neg\langle b \rangle u$. This defines an embedding of LTS into $\mathcal{L}$-expressions whose image are precisely the $\mathcal{L}$-implementations.

Definition 8. Let $\mathcal{E}_1 = (X_1, X_1^0, \Phi_1)$ and $\mathcal{E}_2 = (X_2, X_2^0, \Phi_2)$ be $\mathcal{L}$-expressions. A relation $R \subseteq X_1 \times X_2$ is a *modal refinement* if it holds for all $(x_1, x_2) \in R$ and all $M_1 \in (\!|\Phi_1(x_1)|\!)$ that there exists $M_2 \in (\!|\Phi_2(x_2)|\!)$ such that

- $\forall (a, y_1) \in M_1 : \exists (a, y_2) \in M_2 : (y_1, y_2) \in R$,
- $\forall (a, y_2) \in M_2 : \exists (a, y_1) \in M_1 : (y_1, y_2) \in R$.

We say that $\mathcal{E}_1$ *modally refines* $\mathcal{E}_2$, denoted $\mathcal{E}_1 \leq_{\mathsf{m}} \mathcal{E}_2$, whenever there exists a modal refinement R such that for all $x_1^0 \in X_1^0$, there exists $x_2^0 \in X_2^0$ for which $(x_1^0, x_2^0) \in R$.

3. Structural Equivalence

We proceed to show that our four specification formalisms are structurally equivalent. To this end, we shall expose six translations between them, see Fig. 5. Section 3.1 is concerned with *dn* and *nd*, Section 3.2 with *al* and *la*, and Section 3.3 with *da* and *ad*. We show in Theorems 9, 11 and 12 that all six translations preserve and reflect modal refinement.

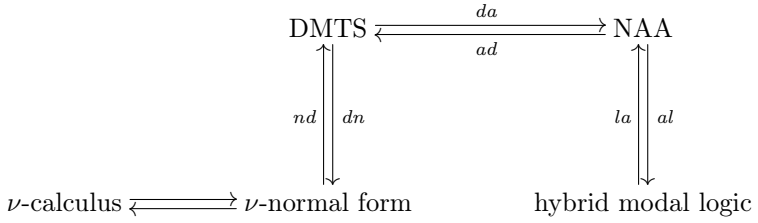

Figure 5: Six translations between specification formalisms

$$
\begin{aligned}
X_1 &= \langle a\rangle Y && \wedge [a]\mathbf{tt} \wedge [b]\mathbf{ff}\\
X_2 &= && [a]\mathbf{ff} \wedge [b]\mathbf{tt}\\
Y &= \langle b\rangle(X_1 \vee X_2) \wedge [a]\mathbf{ff} \wedge [b]\mathbf{tt}
\end{aligned}
$$

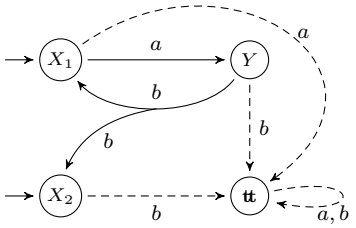

Figure 6: ν-calculus expression in normal form and its DMTS translation, *cf.* Example 2. The state corresponding to **ff** is inconsistent and not shown

3.1. DMTS vs. the Modal ν-Calculus

Our first two translations are rather straight-forward. For a DMTS $\mathcal{D} = (S, S^0, \dashrightarrow, \longrightarrow)$ and all $s \in S$, define $\Diamond(s) = \{N \mid s \longrightarrow N\}$ and, for each $a \in \Sigma$, $\Box^a(s) = \{t \mid s \overset{a}{\dashrightarrow} t\}$. Then, let

$$\Delta(s) = \bigwedge_{N \in \Diamond(s)} \Big(\bigvee_{(a,t)\in N} \langle a\rangle t \Big) \wedge \bigwedge_{a\in\Sigma} [a] \Big(\bigvee_{t \in \Box^a(s)} t \Big) \tag{3}$$

and define the (normal-form) ν-calculus expression $dn(\mathcal{D}) = (S, S^0, \Delta)$.

Note how the formula precisely expresses that we demand at least one of every choice of disjunctive must-transitions (first part) and permit all may-transitions (second part); this is similar to the *characteristic formulae* of [33].

Conversely, for a ν-calculus expression $\mathcal{N} = (X, X^0, \Delta)$ in normal form, let

$$
\begin{aligned}
\dashrightarrow &= \{(x, a, y) \in X \times \Sigma \times X \mid y \in \Box^a(x)\},\\
\longrightarrow &= \{(x, N) \mid x \in X, N \in \Diamond(x)\}.
\end{aligned}
$$

and define the DMTS $nd(\mathcal{N}) = (X, X^0, \dashrightarrow, \longrightarrow)$. Note how this is a simple syntactic translation from boxes to disjunctive must-transitions and from diamonds to may-transitions. Also, the two translations are inverse to each other: $dn(nd(\mathcal{N})) = \mathcal{N}$ and $nd(dn(\mathcal{D})) = \mathcal{D}$. The following theorem follows easily:

Example 2. Consider the ν-calculus formula

$$X = \big(\langle a\rangle(\langle b\rangle X \wedge [a]\mathbf{ff}) \wedge [b]\mathbf{ff}\big) \vee [a]\mathbf{ff}\,.$$

Converting the formula into the normal form of Lemma 4 yields the result given in Fig. 6 (left), where both X_1 and X_2 are initial variables. The resulting DMTS is illustrated in Fig. 6 (right). □

Theorem 9. *For all DMTS $\mathcal{D}_1$, $\mathcal{D}_2$, $\mathcal{D}_1 \leq_{\mathsf{m}} \mathcal{D}_2$ iff $dn(\mathcal{D}_1) \leq_{\mathsf{m}} dn(\mathcal{D}_2)$. For all ν-calculus expressions $\mathcal{N}_1$, $\mathcal{N}_2$ in normal form, $\mathcal{N}_1 \leq_{\mathsf{m}} \mathcal{N}_2$ iff $nd(\mathcal{N}_1) \leq_{\mathsf{m}} nd(\mathcal{N}_2)$.* □

As a corollary, we can now show that the fixed-point semantics and our refinement semantics for the modal ν-calculus agree:

Theorem 10. *For any LTS $\mathcal{I}$ and any ν-calculus expression $\mathcal{N}$ in normal form, $\mathcal{I} \models \mathcal{N}$ iff $\mathcal{I} \leq_{\mathsf{m}} \mathcal{N}$.*

Proof. We show that $\mathcal{I} \leq_{\mathsf{m}} \mathcal{D}$ iff $\mathcal{I} \models dn(\mathcal{D})$ for any DMTS $\mathcal{D}$; the claim then follows because $\mathcal{I} \leq_{\mathsf{m}} \mathcal{N}$ iff $\mathcal{I} \leq_{\mathsf{m}} nd(\mathcal{N})$ iff $\mathcal{I} \models dn(nd(\mathcal{N})) = \mathcal{N}$.

Write $\mathcal{I} = (I, i^0, \longrightarrow_I)$, $\mathcal{D} = (S, S^0, \dashrightarrow, \longrightarrow)$, and $dn(\mathcal{D}) = (S, S^0, \Delta)$.

For states $i \in I$, $s \in S$, write $i \leq_{\mathsf{m}} s$ iff $(I, i, \longrightarrow_I) \in [\![(S, \{s\}, \dashrightarrow, \longrightarrow)]\!]$, *i.e.* if the LTS $\mathcal{I}$ with its initial state replaced by i implements the DMTS S with initial state s. Similarly, write $i \models s$ iff $(I, i, \longrightarrow_I) \in [\![(S, \{s\}, \Delta)]\!]$.

We start with the only-if part. The proof is done by coinduction. We define the assignment $\sigma : S \to 2^I$ as follows: $\sigma(t) = \{j \in I \mid j \leq_{\mathsf{m}} t\}$. We need to show that for every $s \in S$, $\sigma(s) \subseteq (\!|\Delta(s)|\!)\sigma$. Let $i \in \sigma(s)$.

As $i \leq_{\mathsf{m}} s$, we know that (1) $\forall s \longrightarrow N : \exists i \xrightarrow{a}_I j, (a,t) \in N : j \leq_{\mathsf{m}} t$ and (2) $\forall i \xrightarrow{a}_I j : \exists s \overset{a}{\dashrightarrow} t : j \leq_{\mathsf{m}} t$.

Due to (1), we see that for all $N \in \Diamond(s)$, there is $i \xrightarrow{a}_I j$ and $(a,t) \in N$ such that $j \in \sigma(t)$ and $i \in (\!|\langle a \rangle t|\!)\sigma$. Hence $i \in \bigwedge_{N \in \Diamond(s)} (\bigvee_{(a,t) \in N} (\!|\langle a \rangle t|\!)\sigma = (\!|\bigwedge_{N \in \Diamond(s)} (\bigvee_{(a,t) \in N} \langle a \rangle t)|\!)\sigma$.

Due to (2), it holds that for every $a \in \Sigma$ and every $i \xrightarrow{a}_I j$, there is $t \in \Box^a(s)$ such that $j \in \sigma(t) \subseteq [\![\bigvee_{t \in \Box^a(s)} t]\!]\sigma$. Hence $i \in \bigwedge_{a \in \Sigma} (\!|[a](\bigvee_{t \in \Box^a(s)} t)|\!)\sigma = (\!|\bigwedge_{a \in \Sigma} [a](\bigvee_{t \in \Box^a(s)} t)|\!)\sigma$. Altogether, we have shown that $i \in (\!|\Delta(s)|\!)$.

Clearly, there is $s^0 \in S^0$ such that $i^0 \in \sigma(s^0)$. Therefore, $\mathcal{I} \models dn(\mathcal{D})$.

For the other direction, define a relation $R \subseteq I \times S$ by $R = \{(j,t) \mid j \models t\}$. We show that R satisfies the conditions of modal refinement.

Let $(i,s) \in R$. As $i \models s$, we know that (1) $\forall N \in \Diamond(s) : \exists (a,t) \in N : i \models \langle a \rangle t$ and (2) $\forall a \in \Sigma : i \models [a](\bigvee_{t \in \Box^a(s)} t)$.

By (1), we know that for all $s \longrightarrow N$, there is $(a,t) \in N$ and $i \xrightarrow{a}_I j$ such that $j \models t$. By (2), it holds that for all $i \xrightarrow{a}_I j$, there is $s \overset{a}{\dashrightarrow} t$ so that $j \models t$. We have shown that $i \leq_{\mathsf{m}} s$.

Clearly, there is $s^0 \in S^0$ for which $(i^0, s^0) \in R$, hence $\mathcal{I} \leq_{\mathsf{m}} \mathcal{D}$. □

3.2. NAA vs. Hybrid Modal Logic

Also the translations between NAA and our hybrid modal logic are straightforward. For an NAA $\mathcal{A} = (S, S^0, \mathrm{Tran})$ and all $s \in S$, let

$$\Phi(s) = \bigvee_{M \in \mathrm{Tran}(s)} \Big(\bigwedge_{(a,t) \in M} \langle a \rangle t \wedge \bigwedge_{(b,u) \notin M} \neg \langle b \rangle u \Big)$$

and define the $\mathcal{L}$-expression $al(\mathcal{A}) = (S, S^0, \Phi)$.

For an $\mathcal{L}$-expression $\mathcal{E} = (X, X^0, \Phi)$ and all $x \in X$, let $\mathrm{Tran}(x) = (\!|\Phi(x)|\!)$ and define the NAA $la(\mathcal{E}) = (X, X^0, \mathrm{Tran})$.

Theorem 11. *For all NAA $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_1 \leq_{\mathsf{m}} \mathcal{A}_2$ iff $al(\mathcal{A}_1) \leq_{\mathsf{m}} al(\mathcal{A}_2)$. For all $\mathcal{L}$-expressions $\mathcal{E}_1$, $\mathcal{E}_2$, $\mathcal{E}_1 \leq_{\mathsf{m}} \mathcal{E}_2$ iff $la(\mathcal{E}_1) \leq_{\mathsf{m}} la(\mathcal{E}_2)$.*

PROOF. We show that for any NAA (S, S^0, Tran) and any $\mathcal{L}$-expression (S, S^0, Φ), $\mathrm{Tran}(s) = (\!|\Phi(s)|\!)$ for every $s \in S$, for both translations. For the second one, *la*, this is clear by definition, and for the first,

$$\begin{aligned}
(\!|\Phi(s)|\!) &= (\!| \bigvee_{M \in \mathrm{Tran}(s)} \big(\bigwedge_{(a,t)\in M} \langle a\rangle t \wedge \bigwedge_{(b,u)\notin M} \neg\langle b\rangle u\big)|\!) \\
&= \bigcup_{M\in\mathrm{Tran}(s)} \big(\bigcap_{(a,t)\in M} \{M' \mid (a,t) \in M'\} \cap \bigcap_{(b,u)\notin M} \{M' \mid (b,u) \notin M'\}\big) \\
&= \bigcup_{M\in\mathrm{Tran}(s)} \big(\{M' \mid \forall (a,t) \in M : (a,t) \in M'\} \\
&\qquad\qquad \cap \{M' \mid \forall (b,u) \notin M : (b,u) \notin M'\}\big) \\
&= \bigcup_{M\in\mathrm{Tran}(s)} \big(\{M' \mid M \subseteq M'\} \cap \{M' \mid M' \subseteq M\}\big) \\
&= \bigcup_{M\in\mathrm{Tran}(s)} M = \mathrm{Tran}(s)
\end{aligned}$$

as was to be shown. □

3.3. DMTS vs. NAA

The translations between DMTS and NAA are somewhat more intricate. For a DMTS $\mathcal{D} = (S, S^0, \dashrightarrow, \longrightarrow)$ and all $s \in S$, let

$$\mathrm{Tran}(s) = \{M \subseteq \Sigma \times S \mid \forall (a,t) \in M : s \overset{a}{\dashrightarrow} t, \forall s \longrightarrow N : N \cap M \neq \emptyset\}$$

and define the NAA $da(\mathcal{D}) = (S, S^0, \mathrm{Tran})$.[2]

For an NAA $\mathcal{A} = (S, S^0, \mathrm{Tran})$, define the DMTS $ad(\mathcal{A}) = (D, D^0, \dashrightarrow, \longrightarrow)$ as follows:

$$\begin{aligned}
D &= \{M \in \mathrm{Tran}(s) \mid s \in S\} \\
D^0 &= \{M^0 \in \mathrm{Tran}(s^0) \mid s^0 \in S^0\} \\
\longrightarrow &= \big\{\big(M, \{(a, M') \mid M' \in \mathrm{Tran}(t)\}\big) \;\big|\; (a,t) \in M\big\} \\
\dashrightarrow &= \{(M, a, M') \mid \exists M \longrightarrow N : (a, M') \in N\}
\end{aligned}$$

Note that the state spaces of $\mathcal{A}$ and $ad(\mathcal{A})$ are not the same; the one of $ad(\mathcal{A})$ may be exponentially larger.

Theorem 12. *For all DMTS $\mathcal{D}_1$, $\mathcal{D}_2$, $\mathcal{D}_1 \leq_{\mathsf{m}} \mathcal{D}_2$ iff $da(\mathcal{D}_1) \leq_{\mathsf{m}} da(\mathcal{D}_2)$. For all NAA $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_1 \leq_{\mathsf{m}} \mathcal{A}_2$ iff $ad(\mathcal{A}_1) \leq_{\mathsf{m}} ad(\mathcal{A}_2)$.*

PROOF. To show that $\mathcal{D}_1 \leq_{\mathsf{m}} \mathcal{D}_2$ implies $da(\mathcal{D}_1) \leq_{\mathsf{m}} da(\mathcal{D}_2)$, write $\mathcal{D}_1 = (S_1, S_1^0, \dashrightarrow_1, \longrightarrow_1)$, $\mathcal{D}_2 = (S_2, S_2^0, \dashrightarrow_2, \longrightarrow_2)$. We have a modal refinement relation (in the DMTS sense) $R \subseteq S_1 \times S_2$. Now let $(s_1, s_2) \in R$ and $M_1 \in \mathrm{Tran}_1(s_1)$, and define

$$M_2 = \{(a, t_2) \mid s_2 \overset{a}{\dashrightarrow}_2 t_2, \exists (a, t_1) \in M_1 : (t_1, t_2) \in R\}.$$

[2]Note that there is an error in the corresponding formula in [5].

The condition

$$\forall(a,t_2) \in M_2 : \exists(a,t_1) \in M_1 : (t_1,t_2) \in R$$

in the definition of NAA refinement is satisfied by construction. For the inverse condition, let $(a,t_1) \in M_1$, then $s_1 \overset{a}{\dashrightarrow}_1 t_1$, so by DMTS refinement, there is $t_2 \in S_2$ with $s_2 \overset{a}{\dashrightarrow}_2 t_2$ and $(t_1,t_2) \in R$, whence $(a,t_2) \in M_2$ by construction.

We are left with showing that $M_2 \in \mathrm{Tran}_2(s_2)$. First we notice that by construction, indeed $s_2 \overset{a}{\dashrightarrow}_2 t_2$ for all $(a,t_2) \in M_2$. Now let $s_2 \longrightarrow_2 N_2$; we need to show that $N_2 \cap M_2 \neq \emptyset$.

By DMTS refinement, we have $s_1 \longrightarrow_1 N_1$ such that $\forall(a,t_1) \in N_1 : \exists(a,t_2) \in N_2 : (t_1,t_2) \in R$. We know that $N_1 \cap M_1 \neq \emptyset$, so let $(a,t_1) \in N_1 \cap M_1$. Then there also is $(a,t_2) \in N_2$ with $(t_1,t_2) \in R$. But $(a,t_2) \in N_2$ implies $s_2 \overset{a}{\dashrightarrow}_2 t_2$, hence $(a,t_2) \in M_2$.

To prove that $da(\mathcal{D}_1) \leq_{\mathsf{m}} da(\mathcal{D}_2)$ implies $\mathcal{D}_1 \leq_{\mathsf{m}} \mathcal{D}_2$, let $R \subseteq S_1 \times S_2$ be a modal refinement relation in the NAA sense and $(s_1,s_2) \in R$.

Let $s_1 \overset{a}{\dashrightarrow}_1 t_1$, then we cannot have $s_1 \longrightarrow_1 \emptyset$. Let $M_1 = \{(a,t_1)\} \cup \bigcup\{N_1 \mid s_1 \longrightarrow_1 N_1\}$, then $M_1 \in \mathrm{Tran}_1(s_1)$ by construction. This implies that there is $M_2 \in \mathrm{Tran}_2(s_2)$ and $(a,t_2) \in M_2$ with $(t_1,t_2) \in R$, but then also $s_2 \overset{a}{\dashrightarrow}_2 t_2$ as was to be shown.

Let $s_2 \longrightarrow_2 N_2$ and assume, for the sake of contradiction, that there is no $s_1 \longrightarrow_1 N_1$ for which $\forall(a,t_1) \in N_1 : \exists(a,t_2) \in N_2 : (t_1,t_2) \in R$ holds. Then for each $s_1 \longrightarrow_1 N_1$, there is an element $(a_{N_1},t_{N_1}) \in N_1$ for which there is no $(a_{N_1},t_2) \in N_2$ with $(t_{N_1},t_2) \in R$.

Let $M_1 = \{(a_{N_1},t_{N_1}) \mid s_1 \longrightarrow_1 N_1\}$, then $M_1 \in \mathrm{Tran}_1(s_1)$ by construction. Hence we have $M_2 \in \mathrm{Tran}_2(s_2)$ satisfying the conditions in the definition of NAA refinement. By construction of $\mathrm{Tran}_2(s_2)$, $N_2 \cap M_2 \neq \emptyset$, so let $(a,t_2) \in N_2 \cap M_2$. Then there exists $(a,t_1) \in M_1$ for which $(t_1,t_2) \in R$, in contradiction to the definition of M_1.

For the proof that $\mathcal{A}_1 \leq_{\mathsf{m}} \mathcal{A}_2$ implies $ad(\mathcal{A}_1) \leq_{\mathsf{m}} ad(\mathcal{A}_2)$, write $\mathcal{A}_1 = (S_1, S_1^0, \mathrm{Tran}_1)$, $\mathcal{A}_2 = (S_2, S_2^0, \mathrm{Tran}_2)$, with DMTS translations $(D_1, D_1^0, \longrightarrow_1, \dashrightarrow_1)$, $(D_2, D_2^0, \longrightarrow_2, \dashrightarrow_2)$. We have a modal refinement relation (in the NAA sense) $R \subseteq S_1 \times S_2$. Define $R' \subseteq D_1 \times D_2$ by

$$\begin{aligned} R' = \{(M_1,M_2) \mid \exists(s_1,s_2) \in R : M_1 \in \mathrm{Tran}_1(s_1), M_2 \in \mathrm{Tran}(s_2), \\ \forall(a,t_1) \in M_1 : \exists(a,t_2) \in M_2 : (t_1,t_2) \in R, \\ \forall(a,t_2) \in M_2 : \exists(a,t_1) \in M_1 : (t_1,t_2) \in R\}. \end{aligned}$$

We show that R' is a modal refinement in the DMTS sense. Let $(M_1,M_2) \in R'$.

Let $M_2 \longrightarrow_2 N_2$. By construction of $\longrightarrow_2$, there is $(a,t_2) \in M_2$ such that $N_2 = \{(a,M_2') \mid M_2' \in \mathrm{Tran}_2(t_2)\}$. Then $(M_1,M_2) \in R'$ implies that there must be $(a,t_1) \in M_1$ for which $(t_1,t_2) \in R$, and we can define $N_1 = \{(a,M_1') \mid M_1' \in \mathrm{Tran}_1(t_1)\}$, whence $M_1 \longrightarrow_1 N_1$.

We show that $\forall(a,M_1') \in N_1 : \exists(a,M_2') \in N_2 : (M_1',M_2') \in R'$. Let $(a,M_1') \in N_1$, then $M_1' \in \mathrm{Tran}_1(t_1)$. From $(t_1,t_2) \in R$ we hence get $M_2' \in \mathrm{Tran}_2(t_2)$, and then $(a,M_2') \in N_2$ by construction of N_2 and $(M_1',M_2') \in R'$ due to the conditions of NAA refinement (applied to $(t_1,t_2) \in R$).

Let $M_1 \overset{a}{\dashrightarrow}_1 M_1'$, then we have $M_1 \longrightarrow_1 N_1$ for which $(a,M_1') \in N_1$ by construction of $\dashrightarrow_1$. This in turn implies that there must be $(a,t_1) \in M_1$ such

that $N_1 = \{(a, M_1'') \mid M_1'' \in \mathrm{Tran}_1(t_1)\}$, and then by $(M_1, M_2) \in R'$, we get $(a, t_2) \in M_2$ for which $(t_1, t_2) \in R$. Let $N_2 = \{(a, M_2') \mid M_2' \in \mathrm{Tran}_2(t_2)\}$, then $M_2 \longrightarrow_2 N_2$ and hence $M_2 \overset{a}{\dashrightarrow}_2 M_2'$ for all $(a, M_2') \in N_2$. On the other hand, the argument in the previous paragraph shows that there is $(a, M_2') \in N_2$ for which $(M_1', M_2') \in R'$.

We miss to show that R' is initialised. Let $M_1^0 \in D_1^0$, then we have $s_1^0 \in S_1^0$ with $M_1^0 \in \mathrm{Tran}_1(s_1^0)$. As R is initialised, this entails that there is $s_2^0 \in S_2^0$ with $(s_1^0, s_2^0) \in R$, which gives us $M_2^0 \in \mathrm{Tran}_2(s_2^0)$ which satisfies the NAA refinement conditions, whence $(M_1^0, M_2^0) \in R'$.

To see that $ad(\mathcal{A}_1) \leq_{\mathsf{m}} ad(\mathcal{A}_2)$ implies $\mathcal{A}_1 \leq_{\mathsf{m}} \mathcal{A}_2$, let $R \subseteq D_1 \times D_2$ be a modal refinement relation in the DMTS sense and define $R' \subseteq S_1 \times S_2$ by

$$R' = \{(s_1, s_2) \mid \forall M_1 \in \mathrm{Tran}_1(s_1) : \exists M_2 \in \mathrm{Tran}_2(s_2) : (M_1, M_2) \in R\}\,;$$

we will show that R' is an NAA modal refinement.

Let $(s_1, s_2) \in R'$ and $M_1 \in \mathrm{Tran}_1(s_1)$, then by construction of R', we have $M_2 \in \mathrm{Tran}_2(s_2)$ with $(M_1, M_2) \in R$.

Let $(a, t_2) \in M_2$ and define $N_2 = \{(a, M_2') \mid M_2' \in \mathrm{Tran}_2(t_2)\}$, then $M_2 \longrightarrow_2 N_2$. Now $(M_1, M_2) \in R$ implies that there must be $M_1 \longrightarrow_1 N_1$ satisfying $\forall (a, M_1') \in N_1 : \exists (a, M_2') \in N_2 : (M_1', M_2') \in R$. We have $(a, t_1) \in M_1$ such that $N_1 = \{(a, M_1') \mid M_1' \in \mathrm{Tran}_1(t_1)\}$; we only miss to show that $(t_1, t_2) \in R'$. Let $M_1' \in \mathrm{Tran}_1(t_1)$, then $(a, M_1') \in N_1$, hence there is $(a, M_2') \in N_2$ with $(M_1', M_2') \in R$, but $(a, M_2') \in N_2$ also entails $M_2' \in \mathrm{Tran}_2(t_2)$.

Let $(a, t_1) \in M_1$ and define $N_1 = \{(a, M_1') \mid M_1' \in \mathrm{Tran}_1(t_1)\}$, then $M_1 \longrightarrow_1 N_1$. Now let $(a, M_1') \in N_1$, then $M_1 \overset{a}{\dashrightarrow}_1 M_1'$, hence we have $M_2 \overset{a}{\dashrightarrow}_2 M_2'$ for some $(M_1', M_2') \in R$ by modal refinement. By construction of $\dashrightarrow_2$, this implies that there is $M_2 \longrightarrow_2 N_2$ with $(a, M_2') \in N_2$, and we have $(a, t_2) \in M_2$ for which $N_2 = \{(a, M_2'') \mid M_2'' \in \mathrm{Tran}_2(t_2)\}$. Now if $M_1'' \in \mathrm{Tran}_1(t_1)$, then $(a, M_1'') \in N_1$, hence there is $(a, M_2'') \in N_2$ with $(M_1'', M_2'') \in R$, but $(a, M_2'') \in N_2$ also gives $M_2'' \in \mathrm{Tran}_2(t_2)$.

We miss to show that R' is initialised. Let $s_1^0 \in S_1^0$, then $\mathrm{Tran}_1(s_1^0) \neq \emptyset$, hence there is $M_1^0 \in \mathrm{Tran}_1(s_1^0)$. As R is initialised, this gets us $M_2^0 \in D_2$ with $(M_1^0, M_2^0) \in R$, but $M_2^0 \in \mathrm{Tran}_2(s_2^0)$ for some $s_2^0 \in S_2^0$, and then $(s_1^0, s_2^0) \in R'$. □

Corollary 13. *For all DMTS $\mathcal{D}$, ν-calculus expressions $\mathcal{N}$, NAA $\mathcal{A}$, and $\mathcal{L}$-expressions $\mathcal{E}$, $dn(\mathcal{D}) \equiv_{\mathsf{t}} da(\mathcal{D}) \equiv_{\mathsf{t}} \mathcal{D}$, $nd(\mathcal{N}) \equiv_{\mathsf{t}} \mathcal{N}$, $ad(\mathcal{A}) \equiv_{\mathsf{t}} al(\mathcal{A}) \equiv_{\mathsf{t}} \mathcal{A}$, and $la(\mathcal{E}) \equiv_{\mathsf{t}} \mathcal{E}$.* □

3.4. Translation Complexity

We have shown that our four specification formalisms are structurally equivalent, which will be useful for us from a theoretical point of view. From a practical point of view however, some of the translations may incur exponential blow-ups, hence care has to be taken. On the other hand, all our translations can be implemented in an on-the-fly manner, only creating states when necessary.

We already noticed that the translation of ν-calculus expressions into normal form may incur an exponential blow-up, so this also affects our translation from the modal ν-calculus to DMTS. When considering only normal-form expressions, the translations to and from DMTS incur no blow-ups.

When translating from NAA to our hybrid modal logic, we see that, due to the complementation $(b, u) \notin M$, the length of a formula $\Phi(s)$ is quadratic

in the representation of $\mathrm{Tran}(s)$. For the reverse translation, the number of Tran constraints can be exponential in the number of states. The worst case is $\Phi(s) = \mathbf{tt}$, which gets translated to $\mathrm{Tran}(s) = 2^{2^{\Sigma \times S}}$.

We note that there is a direct translation from DMTS to hybrid modal logic: for a DMTS $\mathcal{D} = (S, S^0, \dashrightarrow, \longrightarrow)$, define $dl(\mathcal{D}) = (S, S^0, \Phi)$ with

$$\Phi(s) = \bigwedge_{s \longrightarrow N} \bigvee_{(a,t) \in N} \langle a \rangle t \wedge \bigwedge_{s \not\dashrightarrow^a u} \neg \langle a \rangle u$$

for all $s \in S$. This translation is again quadratic, and $da(\mathcal{D}) = la(dl(\mathcal{D}))$.

The translations between DMTS and NAA may involve exponential blow-ups both ways. For the first translation, we can see this by considering the one-state DMTS $(\{s\}, \{s\}, \dashrightarrow, \longrightarrow)$ with $\dashrightarrow = \{(s, a, s) \mid a \in \Sigma\}$ and $\longrightarrow = \emptyset$. Then $\mathrm{Tran}(s) = 2^{2^{\Sigma \times S}}$.

The fact that also the translation from NAA to DMTS may be exponential in space is evident from the definition. To see that this blow-up is unavoidable, we expose a special property of the Tran-sets arising in the DMTS-to-NAA translation.

Lemma 14. *Let $\mathcal{D} = (S, S^0, \dashrightarrow, \longrightarrow)$ be a DMTS and $s \in S$. For all $M_1, M_2 \in \mathrm{Tran}(s)$ and all $M \subseteq \Sigma \times S$ with $M_1 \subseteq M \subseteq M_1 \cup M_2$, also $M \in \mathrm{Tran}(s)$.*

Proof. For $i = 1, 2$, since $M_i \in \mathrm{Tran}(s)$, we know that

- for all $(a, t) \in M_i$, $s \overset{a}{\dashrightarrow} t$, and
- for all $s \longrightarrow N$, there is $(a, t) \in M_i \cap N$.

Now as $M \subseteq M_1 \cup M_2$, it directly follows that for all $(a, t) \in M$, we have $s \overset{a}{\dashrightarrow} t$. Moreover, since $M_1 \subseteq M$, we also have that for all $s \longrightarrow N$, there exists $(a, t) \in M \cap N$. As a consequence, $M \in \mathrm{Tran}(s)$. □

Using this, we can show the following.[3]

Lemma 15. *There exists a one-state NAA $\mathcal{A}$ for which any DMTS $\mathcal{D} \equiv_{\mathsf{t}} \mathcal{A}$ has at least 2^{n-1} states, where n is the size of the alphabet Σ.*

Proof. Let $\Sigma = \{a_1, \ldots, a_n\}$ and $\mathcal{A} = (\{s^0\}, \{s^0\}, \mathrm{Tran})$ the NAA with $\mathrm{Tran}(s^0) = \{M \subseteq \Sigma \times \{s^0\} \mid \exists k : |M| = 2k\}$ the transition constraint containing all disjunctive choices of even cardinality. Let $\mathcal{D} = (T, T^0, \dashrightarrow, \longrightarrow)$ be a DMTS with $\mathcal{D} \equiv_{\mathsf{t}} \mathcal{A}$; we claim that $\mathcal{D}$ must have at least 2^{n-1} initial states.

Assume, for the purpose of contradiction, that $T^0 = \{t_1^0, \ldots, t_m^0\}$ with $m < 2^{n-1}$. We must have $\bigcup_{i=1}^{m} \mathrm{Tran}_T(t_i^0) = \{M \subseteq \Sigma \times T \mid \exists k : |M| = 2k\}$, so that there is an index $j \in \{1, \ldots, m\}$ for which $\mathrm{Tran}_T(t_j^0) = \{M_1, M_2\}$ contains two different disjunctive choices from $\mathrm{Tran}_S(s^0)$. By Lemma 14, also $M \in \mathrm{Tran}_T(t_j^0)$ for any M with $M_1 \subseteq M \subseteq M_1 \cup M_2$. But $M_1 \cup M_2$ has greater cardinality than M_1, so that there will be an $M \in \mathrm{Tran}_T(t_j^0)$ with odd cardinality. □

Figure 7 sums up the translation complexities.

[3] We wish to thank Ilya Bogdanov for discussions on this subject [38].

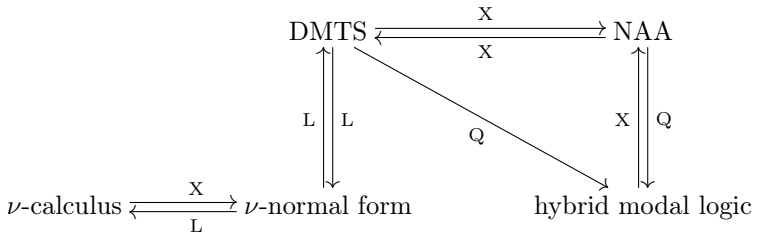

Figure 7: Complexity of the translations between specification formalisms. "L" stands for linear (no blow-up), "Q" for quadratic blow-up, and "X" for exponential blow-up

3.5. Initial States

We finish this section with a justification for why we allow our specifications to have several (or possibly zero) initial states. The first lemma shows that for NAA, and up to *thorough* refinement, this is inessential; due to their close relationship, this also holds for $\mathcal{L}$-expressions.

Lemma 16. *For any NAA $\mathcal{A}_1$, there is an NAA $\mathcal{A}_2 = (S_2, S_2^0, \mathrm{Tran}_2)$ with $S_2^0 = \{s_2^0\}$ a singleton and $\mathcal{A}_1 \equiv_{\mathsf{t}} \mathcal{A}_2$.*

Proof. Write $\mathcal{A}_1 = (S_1, S_1^0, \mathrm{Tran}_1)$. If $S_1^0 = \emptyset$, we can let $\mathcal{S}_2 = \{s_2^0\}$ and $\mathrm{Tran}_2(s_2^0) = \emptyset$. Otherwise, we let $S_2 = S_1 \cup \{s_2^0\}$, where s_2^0 is a new state, and $\mathrm{Tran}_2(s) = \mathrm{Tran}_1(s)$ for $s \in S_1$, $\mathrm{Tran}_2(s_2^0) = \bigcup_{s_1^0 \in S_1^0} \mathrm{Tran}_1(s_1^0)$. Let $R = \mathrm{id}_{S_1} \cup \{(s_1^0, s_2^0) \mid s_1^0 \in S_1^0\}$, then R is easily seen to be a modal refinement $\mathcal{A}_1 \leq_{\mathsf{m}} \mathcal{A}_2$.

We show that $\mathcal{A}_2 \leq_{\mathsf{t}} \mathcal{A}_1$. Let $\mathcal{I} = (S, s^0, \longrightarrow) \in [\![\mathcal{A}_2]\!]$, then we have a modal refinement $R_2 \subseteq S \times S_2$, *i.e.* such that for all $(s, s_2) \in R_2$, there exists $M_2 \in \mathrm{Tran}_2(s_2)$ for which

$$\begin{aligned} &\forall s \xrightarrow{a} t : \exists (a, t_2) \in M_2 : (t, t_2) \in R_2 \,, \\ &\forall (a, t_2) \in M_2 : \exists s \xrightarrow{a} t : (t, t_2) \in R_2 \,. \end{aligned} \tag{4}$$

Now $(s^0, s_2^0) \in R_2$ implies that there must be $M_2 \in \mathrm{Tran}_2(s_2^0)$ for which (4) holds, but by definition of $\mathrm{Tran}_2(s_2^0)$, this entails that there is $s_1^0 \in S_1^0$ for which $M_2 \in \mathrm{Tran}_1(s_1^0)$. Define $R_1 \subseteq S \times S_1$ by

$$R_1 = \{(s, s_2) \mid (s, s_2) \in R_2, s \neq s^0\} \cup \{(s^0, s_1^0)\} \,,$$

then R_1 is a modal refinement $\mathcal{I} \leq_{\mathsf{m}} \mathcal{A}_1$. □

In order to show that the above statement does *not* hold for DMTS, we expose a special property of DMTS with single initial states, *cf.* [7, Example 7.8]. Recall that for LTS $\mathcal{I}_1 = (S_1, s_1^0, \longrightarrow_1)$, $\mathcal{I}_2 = (S_2, s_2^0, \longrightarrow_2)$, their *nondeterministic sum* is given by $\mathcal{I}_1 + \mathcal{I}_2 = (S, s^0, \longrightarrow)$ with $S = S_1 \cup S_2 \cup \{s^0\}$ (with the unions disjoint), where s^0 is a new state, and transitions $s \xrightarrow{a} t$ iff $s \xrightarrow{a}_1 t$ or $s \xrightarrow{a}_2 t$ together with $s^0 \xrightarrow{a} t$ for all t with $s_1^0 \xrightarrow{a}_1 t$ or $s_2^0 \xrightarrow{a}_2 t$.

Lemma 17. *If $\mathcal{D} = (S, \{s^0\}, \dashrightarrow, \longrightarrow)$ is a DMTS with a single initial state and $\mathcal{I}_1, \mathcal{I}_2 \in [\![\mathcal{D}]\!]$, then also $\mathcal{I}_1 + \mathcal{I}_2 \in [\![\mathcal{D}]\!]$.*

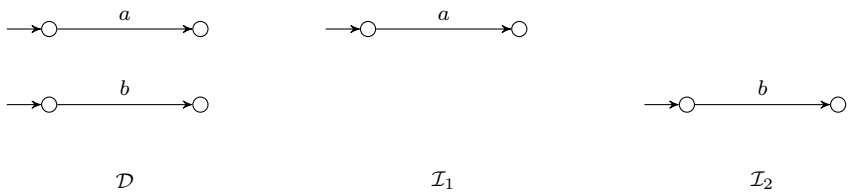

Figure 8: DMTS $\mathcal{D}$ with two initial states and its two implementations $\mathcal{I}_1$, $\mathcal{I}_2$

Proof. Let i_1^0 and i_2^0 be the initial states of $\mathcal{I}_1$ and $\mathcal{I}_2$, respectively. Let further i^0 be the initial state of $\mathcal{I}_1 + \mathcal{I}_2$. Assume that we have modal refinements R_1 and R_2 such that $(i_1^0, s^0) \in R_1$ and $(i_2^0, s^0) \in R_2$. Let $R = R_1 \cup R_2 \cup \{(i^0, s^0)\}$. Clearly, R is a modal refinement witnessing $\mathcal{I}_1 + \mathcal{I}_2 \leq_{\mathsf{m}} \mathcal{D}$. □

Now let $\mathcal{D}$ be the DMTS, with *two* initial states, depicted in Fig. 8, then $[\![\mathcal{D}]\!] = \{\mathcal{I}_1, \mathcal{I}_2\}$ as also seen in Fig. 8, but $\mathcal{I}_1 + \mathcal{I}_2 \notin [\![\mathcal{D}]\!]$. Hence $\mathcal{D}$ is not thoroughly equivalent to any DMTS with a single initial state.

Applying the construction from the proof of Lemma 16 to the DMTS in Fig. 8 gives an NAA $\mathcal{A}_2 = (S_2, \{s_2^0\}, \mathrm{Tran}_2)$ with $\mathrm{Tran}_2(s_2^0) = \{\{(a, s_1)\}, \{(b, t_1\}\}$ (where s_1 and t_1 are the target states of the a and b transitions in $\mathcal{D}$, respectively). This specifies an *exclusive disjunction*: one of a and b has to be implemented, but not both. This also serves to show that Lemma 17 does not hold for NAA.

Corollary 18. *There is a DMTS $\mathcal{D}_1$ for which there is no DMTS $\mathcal{D}_2 = (S_2, S_2^0, \dashrightarrow, \longrightarrow)$ with $S_2^0 = \{s_2^0\}$ a singleton and $\mathcal{D}_1 \equiv_{\mathsf{t}} \mathcal{D}_2$.* □

Due to their close relationship with DMTS, this property also holds for ν-calculus expressions *in normal form*: there exist ν-calculus expressions which are not equivalent to any normal-form ν-calculus expression with a single initial variable. (Of course, omitting "normal form" would make this statement invalid; as disjunction is part of the syntax, any ν-calculus expression is thoroughly equivalent to one with only one initial variable.)

We also remark that the above argument can easily be extended to show that for any $k \in \mathbb{N}$, there exists a DMTS with $k + 1$ initial states which is not thoroughly equivalent to any DMTS with at most k initial states.

Using again the example in Fig. 8, we can also show that the statement in Lemma 16 does *not* hold when thorough equivalence is replaced by modal equivalence. Let $\mathcal{A}_1 = da(\mathcal{D})$, with initial states s^0 and t^0, be the NAA translation of the DMTS in Fig. 8 and assume that there exists an NAA $\mathcal{A}_2$ with single initial state s_2^0 for which $\mathcal{A}_2 \leq_{\mathsf{m}} \mathcal{A}_1$. Then there is a modal refinement R with $(s_2^0, s^0), (s_2^0, t^0) \in R$. Let $M_2 \in \mathrm{Tran}_2(s_2^0)$, then by $(s_2^0, s^0) \in R$, there must be some $(a, t_2) \in M_2$ with $(t_2, s_1) \in R$. By $(s_2^0, t^0) \in R$, this implies that there must be $(a, t_1) \in \mathrm{Tran}_1(t^0)$, a contradiction.

4. Specification Theory

Behavioural specifications typically come equipped with operations which allow for *compositional reasoning*, *viz.* conjunction, composition and quotient, *cf.* [2]. On deterministic MTS, these operations can be given easily using simple

structural operational rules. For non-deterministic systems this is significantly harder.

We remark that composition and quotient operators are well-known from some logics, such as, *e.g.* linear [25] or spatial logic [15], and were extended to quite general contexts [16]. However, whereas these operators are part of the formal syntax in those logics, for us they are simply operations on logical expressions (or DMTS, or NAA). Consequently, composition is generally only a sound over-approximation of the semantic composition.

Given the structural equivalence of DMTS, the modal ν-calculus, NAA, and our hybrid modal logic exposed in the previous section, it suffices to introduce the operations for *one* of the four types of specifications. On the other hand, we will often state properties for *all* four types of specifications at the same time, letting $\mathcal{S}$ stand for a specification of any type.

4.1. Disjunction and Conjunction

Disjunction of specifications is easily defined as we allow multiple initial states. For DMTS $\mathcal{D}_1 = (S_1, S_1^0, \dashrightarrow_1, \longrightarrow_1)$, $\mathcal{D}_2 = (S_2, S_2^0, \dashrightarrow_2, \longrightarrow_2)$, we can hence define $\mathcal{D}_1 \vee \mathcal{D}_2 = (S_1 \cup S_2, S_1^0 \cup S_2^0, \dashrightarrow_1 \cup \dashrightarrow_2, \longrightarrow_1 \cup \longrightarrow_2)$ (with all unions disjoint). Similar definitions are available for the other types of specifications, and disjunction commutes with the translations.

Conjunction for DMTS is an extension of the construction from [4] for multiple initial states. Given two DMTS $(S_1, S_1^0, \dashrightarrow_1, \longrightarrow_1)$, $(S_2, S_2^0, \dashrightarrow_2, \longrightarrow_2)$, we define $S_1 \wedge S_2 = (S, S^0, \dashrightarrow, \longrightarrow)$ with $S = S_1 \times S_2$, $S^0 = S_1^0 \times S_2^0$, and

- $(s_1, s_2) \overset{a}{\dashrightarrow} (t_1, t_2)$ iff $s_1 \overset{a}{\dashrightarrow}_1 t_1$ and $s_2 \overset{a}{\dashrightarrow}_2 t_2$,
- for all $s_1 \longrightarrow_1 N_1$, $(s_1, s_2) \longrightarrow \{(a, (t_1, t_2)) \mid (a, t_1) \in N_1, (s_1, s_2) \overset{a}{\dashrightarrow} (t_1, t_2)\}$,
- for all $s_2 \longrightarrow_2 N_2$, $(s_1, s_2) \longrightarrow \{(a, (t_1, t_2)) \mid (a, t_2) \in N_2, (s_1, s_2) \overset{a}{\dashrightarrow} (t_1, t_2)\}$.

For NAA, conjunction can be defined using auxiliary projection functions $\pi_i : \Sigma \times S_1 \times S_2 \to \Sigma \times S_i$ given by

$$\pi_1(M) = \{(a, s_1) \mid \exists s_2 \in S_2 : (a, s_1, s_2) \in M\},$$
$$\pi_2(M) = \{(a, s_2) \mid \exists s_1 \in S_1 : (a, s_1, s_2) \in M\}.$$

Then for NAA $\mathcal{A}_1 = (S_1, S_1^0, \mathrm{Tran}_1)$, $\mathcal{A}_2 = (S_2, S_2^0, \mathrm{Tran}_2)$, we let $\mathcal{A}_1 \wedge \mathcal{A}_2 = (S, S^0, \mathrm{Tran})$, with $S = S_1 \times S_2$, $S^0 = S_1^0 \times S_2^0$ and $\mathrm{Tran}((s_1, s_2)) = \{M \subseteq \Sigma \times S_1 \times S_2 \mid \pi_1(M) \in \mathrm{Tran}_1(s_1), \pi_2(M) \in \mathrm{Tran}_2(s_2)\}$.

We can also define conjunction for $\mathcal{L}$-expressions, using similar auxiliary mappings on formulae. For sets X_1, X_2 and $i \in \{1, 2\}$, we define $\rho_i : \mathcal{L}(X_i) \to \mathcal{L}(X_1 \times X_2)$ inductively, by

- $\rho_i(\mathbf{tt}) = \mathbf{tt}$, $\rho_i(\mathbf{ff}) = \mathbf{ff}$, $\rho_i(\neg\phi) = \neg\rho_i(\phi)$, $\rho_i(\phi_i \wedge \phi_2) = \rho_i(\phi_i) \wedge \rho_i(\phi_2)$,
- $\rho_1(\langle a \rangle x_1) = \bigvee_{x_2 \in X_2} \langle a \rangle (x_1, x_2)$,
- $\rho_2(\langle a \rangle x_2) = \bigvee_{x_1 \in X_1} \langle a \rangle (x_1, x_2)$.

Then for $\mathcal{L}$-expressions (X_1, X_1^0, Φ_1), (X_2, X_2^0, Φ_2), $X_1 \wedge X_2 = (X_1 \times X_2, X_1^0 \times X_2^0, \Phi)$ with $\Phi((x_1, x_2)) = \rho_1(\Phi_1(x_1)) \wedge \rho_2(\Phi_2(x_2))$.

Lemma 19. *For all DMTS $\mathcal{D}_1$, $\mathcal{D}_2$, NAA $\mathcal{A}_1$, $\mathcal{A}_2$, and $\mathcal{L}$-expressions $\mathcal{E}_1$, $\mathcal{E}_2$, $da(\mathcal{D}_1 \wedge \mathcal{D}_2) = da(\mathcal{D}_1) \wedge da(\mathcal{D}_2)$, $ad(\mathcal{A}_1 \wedge \mathcal{A}_2) \equiv_{\mathsf{m}} ad(\mathcal{A}_1) \wedge ad(\mathcal{A}_2)$, $al(\mathcal{A}_1 \wedge \mathcal{A}_2) = al(\mathcal{A}_1) \wedge al(\mathcal{A}_2)$, and $la(\mathcal{E}_1 \wedge \mathcal{E}_2) = la(\mathcal{E}_1) \wedge la(\mathcal{E}_2)$.*

Note that above we only claim modal equivalence, not equality, for the *ad* translation; this is due to the change of state space during the translation and follows easily from Theorem 20 below.

Proof. The last two claims follow easily once one notices that for $i \in \{1, 2\}$ and all M, $M \models \rho_i(\phi)$ iff $\pi_i(M) \models \phi$. To show the first claim, let $\mathcal{D}_1 = (S_1, S_1^0, \dashrightarrow_1, \longrightarrow_1)$ and $\mathcal{D}_2 = (S_2, S_2^0, \dashrightarrow_2, \longrightarrow_2)$ be DMTS, with NAA translations Let $da(\mathcal{D}_1) = (S_1, S_1^0, \mathrm{Tran}_1)$ and $da(\mathcal{D}_2) = (S_2, S_2^0, \mathrm{Tran}_2)$. Write $\mathcal{A}^\wedge = da(\mathcal{D}_1 \wedge \mathcal{D}_2)$ and $\mathcal{A}_\wedge = da(\mathcal{D}_1) \wedge da(\mathcal{D}_2)$; we show that $\mathcal{A}^\wedge = \mathcal{A}_\wedge$.

First, remark that $\mathcal{A}^\wedge$ and $\mathcal{A}_\wedge$ have precisely the same state space $S_1 \times S_2$ and initial states $S_1^0 \times S_2^0$. We now show that they have the same transition constraints. Let $\mathrm{Tran}_\wedge$ (resp. $\mathrm{Tran}^\wedge$) be the transition constraints mapping of $\mathcal{A}_\wedge$ (resp. $\mathcal{A}^\wedge$). Let $(s_1, s_2) \in S_1 \times S_2$ and $M \in \mathrm{Tran}_\wedge(s_1, s_2)$.

By construction of $\mathrm{Tran}_\wedge$, there must be $M_1 \in \mathrm{Tran}_1(s_1)$ and $M_2 \in \mathrm{Tran}_2(s_2)$ such that $\pi_1(M) = M_1$ and $\pi_2(M) = M_2$. We show that $M \in \mathrm{Tran}^\wedge(s_1, s_2)$. Let $(a, (t_1, t_2)) \in M$. Since $\pi_1(M) = M_1$ and $\pi_2(M) = M_2$, we have $(a, t_1) \in M_1$ and $(a, t_2) \in M_2$. As a consequence, there are transitions $s_1 \overset{a}{\dashrightarrow} t_1$ and $s_2 \overset{a}{\dashrightarrow} t_2$ in $\mathcal{D}_1$ and $\mathcal{D}_2$, respectively. Thus, by construction, there is a transition $(s_1, s_2) \overset{a}{\dashrightarrow} (t_1, t_2)$ in $\mathcal{D}_1 \wedge \mathcal{D}_2$.

Let $(s_1, s_2) \longrightarrow N$ in $\mathcal{D}_1 \wedge \mathcal{D}_2$. By construction, N is such that either (1) there exists N_1 such that $s_1 \longrightarrow N_1$ in $\mathcal{D}_1$ and $N = \{(a, (t_1, t_2)) \mid (a, t_1) \in N_1, (s_1, s_2) \overset{a}{\dashrightarrow} (t_1, t_2)\}$, or (2) there exists N_2 such that $s_2 \longrightarrow N_2$ in $\mathcal{D}_2$ and $N = \{(a, (t_1, t_2)) \mid (a, t_2) \in N_2, (s_1, s_2) \overset{a}{\dashrightarrow} (t_1, t_2)\}$. Assume that (1) holds (case (2) being symmetric). Since $M_1 \in \mathrm{Tran}_1(s_1)$, there must be $(a, t_1) \in N_1 \cap M_1$. As $\pi_1(M) = M_1$, there must be $t_2 \in S_2$ such that $(a, (t_1, t_2)) \in M$. As a consequence, there is $(a, (t_1, t_2)) \in M \cap N$.

We have shown that $M \in \mathrm{Tran}^\wedge(s_1, s_2)$. Similarly, we can show that for all $M \in \mathrm{Tran}^\wedge(s_1, s_2)$, we also have $M \in \mathrm{Tran}_\wedge(s_1, s_2)$. We can thus conclude that $\mathrm{Tran}^\wedge = \mathrm{Tran}_\wedge$, hence $\mathcal{A}^\wedge = \mathcal{A}_\wedge$. □

Theorem 20. *For all specifications $\mathcal{S}_1$, $\mathcal{S}_2$, $\mathcal{S}_3$,*

- *$\mathcal{S}_1 \vee \mathcal{S}_2 \leq_{\mathsf{m}} \mathcal{S}_3$ iff $\mathcal{S}_1 \leq_{\mathsf{m}} \mathcal{S}_3$ and $\mathcal{S}_2 \leq_{\mathsf{m}} \mathcal{S}_3$,*
- *$\mathcal{S}_1 \leq_{\mathsf{m}} \mathcal{S}_2 \wedge \mathcal{S}_3$ iff $\mathcal{S}_1 \leq_{\mathsf{m}} \mathcal{S}_2$ and $\mathcal{S}_1 \leq_{\mathsf{m}} \mathcal{S}_3$,*
- *$[\![\mathcal{S}_1 \vee \mathcal{S}_2]\!] = [\![\mathcal{S}_1]\!] \cup [\![\mathcal{S}_2]\!]$, and $[\![\mathcal{S}_1 \wedge \mathcal{S}_2]\!] = [\![\mathcal{S}_1]\!] \cap [\![\mathcal{S}_2]\!]$.*

Proof. The proof that $\mathcal{S}_1 \vee \mathcal{S}_2 \leq_{\mathsf{m}} \mathcal{S}_3$ iff $\mathcal{S}_1 \leq_{\mathsf{m}} \mathcal{S}_3$ and $\mathcal{S}_2 \leq_{\mathsf{m}} \mathcal{S}_3$ is trivial: any modal refinement $R \subseteq (S_1 \cup S_2) \times S_3$ splits into two refinements $R_1 \subseteq S_1 \times S_3$, $R_2 \subseteq S_2 \times S_3$ and vice versa.

For the proof of the second claim, which we show for DMTS, we prove the back direction first. Let $R_2 \subseteq S_1 \times S_2$, $R_3 \subseteq S_1 \times S_3$ be (DMTS) modal refinements and define $R = \{(s_1, (s_2, s_3)) \mid (s_1, s_2) \in R_1, (s_1, s_3) \in R_3\} \subseteq S_1 \times (S_2 \times S_3)$. Then R is initialised.

Now let $(s_1, (s_2, s_3)) \in R$, then $(s_1, s_2) \in R_2$ and $(s_1, s_3) \in R_3$. Assume that $s_1 \overset{a}{\dashrightarrow}_1 t_1$, then by $\mathcal{S}_1 \leq_{\mathsf{m}} \mathcal{S}_2$, we have $s_2 \overset{a}{\dashrightarrow}_2 t_2$ with $(t_1, t_2) \in R_2$. Similarly, by

$\mathcal{S}_1 \leq_{\mathsf{m}} \mathcal{S}_3$, we have $s_3 \overset{a}{\dashrightarrow}_3 t_3$ with $(t_1, t_3) \in R_3$. But then also $(t_1, (t_2, t_3)) \in R$, and $(s_2, s_3) \overset{a}{\dashrightarrow} (t_2, t_3)$ by definition.

Assume that $(s_2, s_3) \longrightarrow N$. Without loss of generality we can assume that there is $s_2 \longrightarrow_2 N_2$ such that $N = \{(a, (t_2, t_3)) \mid (a, t_2) \in N_2, s_3 \overset{a}{\dashrightarrow}_3 t_3\}$. By $S_1 \leq_{\mathsf{m}} S_2$, we have $s_1 \longrightarrow_1 N_1$ such that $\forall (a, t_1) \in N_1 : \exists (a, t_2) \in N_2 : (t_1, t_2) \in R_2$.

Let $(a, t_1) \in N_1$, then also $s_1 \overset{a}{\dashrightarrow}_1 t_1$, so by $S_1 \leq_{\mathsf{m}} S_3$, there is $s_3 \overset{a}{\dashrightarrow}_3 t_3$ with $(t_1, t_3) \in R_3$. By the above, we also have $(a, t_2) \in N_2$ such that $(t_1, t_2) \in R_2$, but then $(a, (t_2, t_3)) \in N$ and $(t_1, (t_2, t_3)) \in R$.

For the other direction of the second claim, let $R \subseteq S_1 \times (S_2 \times S_3)$ be a (DMTS) modal refinement. We show that $S_1 \leq_{\mathsf{m}} S_2$, the proof of $S_1 \leq_{\mathsf{m}} S_3$ being entirely analogous. Define $R_2 = \{(s_1, s_2) \mid \exists s_3 \in S_3 : (s_1, (s_2, s_3)) \in R\} \subseteq S_1 \times S_2$, then R_2 is initialised.

Let $(s_1, s_2) \in R_2$, then we must have $s_3 \in S_3$ such that $(s_1, (s_2, s_3)) \in R$. Assume that $s_1 \overset{a}{\dashrightarrow}_1 t_1$, then also $(s_2, s_3) \overset{a}{\dashrightarrow} (t_2, t_3)$ and $(t_1, (t_2, t_3)) \in R$. By construction we have $s_2 \overset{a}{\dashrightarrow}_2 t_2$ and $s_3 \overset{a}{\dashrightarrow}_3 t_3$, but then $(t_1, t_2) \in R_2$.

Assume that $s_2 \longrightarrow_2 N_2$, then by construction, $(s_2, s_3) \longrightarrow N = \{(a, (t_2, t_3)) \mid (a, t_2) \in N_2, s_3 \overset{a}{\dashrightarrow}_3 t_3\}$. By $\mathcal{S}_1 \leq_{\mathsf{m}} \mathcal{S}_2 \wedge \mathcal{S}_3$, we have $s_1 \longrightarrow_1 N_1$ such that $\forall (a, t_1) \in N_1 : \exists (a, (t_2, t_3)) \in N : (t_1, (t_2, t_3)) \in R$.

Let $(a, t_1) \in N_1$, then we have $(a, (t_2, t_3)) \in N$ for which $(t_1, (t_2, t_3)) \in R$. By construction of N, this implies that there are $(a, t_2) \in N_2$ and $s_3 \overset{a}{\dashrightarrow}_3 t_3$, but then $(t_1, t_2) \in R$.

As to the last claims of the theorem, $[\![\mathcal{S}_1 \wedge \mathcal{S}_2]\!] = [\![\mathcal{S}_1]\!] \cap [\![\mathcal{S}_2]\!]$ is clear from what we just proved: for all implementations $\mathcal{I}$, $\mathcal{I} \leq_{\mathsf{m}} \mathcal{S}_1 \wedge \mathcal{S}_2$ iff $\mathcal{I} \leq_{\mathsf{m}} \mathcal{S}_1$ and $\mathcal{I} \leq_{\mathsf{m}} \mathcal{S}_2$. For the other part, it is clear by construction that for any implementation $\mathcal{I}$, any witness R for $\mathcal{I} \leq_{\mathsf{m}} \mathcal{S}_1$ is also a witness for $\mathcal{I} \leq_{\mathsf{m}} \mathcal{S}_1 \vee \mathcal{S}_2$, and similarly for $\mathcal{S}_2$, hence $[\![\mathcal{S}_1]\!] \cup [\![\mathcal{S}_2]\!] \subseteq [\![\mathcal{S}_1 \vee \mathcal{S}_2]\!]$.

To show the other inclusion, we note that an initialised refinement R witnessing $\mathcal{I} \leq_{\mathsf{m}} \mathcal{S}_1 \vee \mathcal{S}_2$ must relate the initial state of $\mathcal{I}$ either to an initial state of $\mathcal{S}_1$ or to an initial state of $\mathcal{S}_2$. In the first case, and by disjointness, R witnesses $\mathcal{I} \leq_{\mathsf{m}} \mathcal{S}_1$, in the second, $\mathcal{I} \leq_{\mathsf{m}} \mathcal{S}_2$. Note how it is essential here that *implementations* have but *one* initial state; this part of the proof would break down if we were to allow several initial states for implementations. □

Corollary 21. *With operations $\vee$ and $\wedge$, each of our four classes of specifications forms a bounded distributive lattice up to $\equiv_{\mathsf{m}}$.*

Proof. The bottom elements (up to $\equiv_{\mathsf{m}}$) in the lattices are given by specifications with empty initial state sets. The top elements are the DMTS $(\{s^0\}, \{s^0\}, \{(s^0, a, s^0) \mid a \in \Sigma\}, \emptyset)$ and its respective translations.

We miss to verify distributivity. Let $\mathcal{A}_i = (S_i, S_i^0, \mathrm{Tran}_i)$, for $i = 1, 2, 3$, be NAA. The set of variables of both $\mathcal{A}_1 \wedge (\mathcal{A}_2 \vee \mathcal{A}_3)$ and $(\mathcal{A}_1 \wedge \mathcal{A}_2) \vee (\mathcal{A}_1 \wedge \mathcal{A}_3)$ is $S_1 \times (S_2 \cup S_3) = S_1 \times S_2 \cup S_1 \times S_3$, and one easily sees that the identity relation is a two-sided modal refinement. Things are similar for the other distributive law. □

4.2. Composition

The composition operator for a specification theory is to mimic, at specification level, the parallel composition of implementations. That is to say, if $\|$ is a

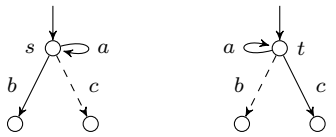

Figure 9: DMTS $\mathcal{S}$ and $\mathcal{T}$ whose composition cannot be captured precisely

composition operator for implementations (LTS), then the goal is to extend $\|$ to specifications such that for all specifications $\mathcal{S}_1$, $\mathcal{S}_2$,

$$[\![\mathcal{S}_1 \| \mathcal{S}_2]\!] = \{\mathcal{I}_1 \| \mathcal{I}_2 \mid \mathcal{I}_1 \in [\![\mathcal{S}_1]\!], \mathcal{I}_2 \in [\![\mathcal{S}_2]\!]\}. \tag{5}$$

For simplicity, we use CSP-style synchronisation for parallel composition of LTS, however, our results readily carry over to other types of composition. Analogously to the situation for MTS [7], we have the following negative result:

Theorem 22. *There is no operator $\|$, for any of our specification formalisms, which satisfies* (5).

Proof. We show that there exist DMTS $\mathcal{S}$ and $\mathcal{T}$ such that there is no DMTS $\mathcal{D}$ with $[\![\mathcal{D}]\!] = [\![\mathcal{S}]\!] \| [\![\mathcal{T}]\!] \coloneqq \{\mathcal{I} \| \mathcal{J} \mid \mathcal{I} \in [\![\mathcal{S}]\!], \mathcal{J} \in [\![\mathcal{T}]\!]\}$. They are given in Figure 9; $\mathcal{S}$ has initial state s, while $\mathcal{T}$ has initial state t. Note that in fact, $\mathcal{S}$ and $\mathcal{T}$ are MTS, *i.e.* no disjunctive must transitions are used.

We make the following observations about implementations of $\mathcal{S}$ and $\mathcal{T}$. They always contain one or more infinite runs labelled with a's with one-step b or c branches. Moreover, all infinite runs in these implementations are of this form. To each infinite a-run of an implementation we assign its signature, that is a word over $2^{\{b,c\}}$ that describes the one-step branches. This means that every implementation of $\mathcal{S}$ has runs with signatures from $\{\{b\},\{b,c\}\}^\omega$, while every implementation of $\mathcal{T}$ has runs with signatures from $\{\{c\},\{b,c\}\}^\omega$.

We now construct an implementation state space as illustrated in Figure 10. Consider the implementations $\mathcal{I}_1, \mathcal{I}_2, \dots$ that share the same state space and have the initial state $i_1, i_2, \dots$, respectively. The implementation $\mathcal{I}_n$ has only one a-run with the signature $\emptyset^n\{b,c\}\emptyset^\omega$. Note that $\mathcal{I}_n$ is the composition of an implementation of $\mathcal{S}$ that has only one a-run with the signature $\{b\}^n\{b,c\}\{b\}^\omega$ and an implementation of $\mathcal{T}$ that has only one a-run with the signature $\{c\}^n\{b,c\}\{c\}^\omega$.

Assume now that there exists a DMTS $\mathcal{D}$ with $[\![\mathcal{D}]\!] = [\![\mathcal{S}]\!] \| [\![\mathcal{T}]\!]$. As all $\mathcal{I}_n$ belong to $[\![\mathcal{D}]\!]$ and there is only a finite number of initial states of $\mathcal{D}$, there has to be at least one initial state of $\mathcal{D}$, say d, such that there exists a modal refinement R containing both (i_k, d) and (i_l, d) for some numbers $k < l$. Let $\mathcal{D}_d$ be created from $\mathcal{D}$ by changing the set of initial states to the singleton $\{d\}$. As both $\mathcal{I}_k \leq_{\mathsf{m}} \mathcal{D}_d$ and $\mathcal{I}_l \leq_{\mathsf{m}} \mathcal{D}_d$ and $\mathcal{D}_d$ has only one initial state, we know by

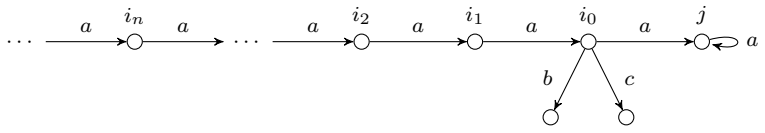

Figure 10: Implementation state space illustrating the proof of Theorem 22

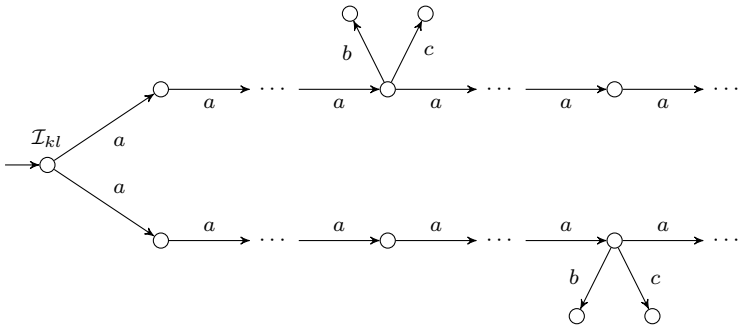

Figure 11: The nondeterministic sum of $\mathcal{I}_k$ and $\mathcal{I}_l$, unfolded

Lemma 17 that also $\mathcal{I}_{kl} = \mathcal{I}_k + \mathcal{I}_l \leq_{\mathsf{m}} \mathcal{D}_d$. The unfolding of this implementation is illustrated in Figure 11.

We now argue that $\mathcal{I}_{kl} \notin [\![\mathcal{S}]\!] \| [\![\mathcal{T}]\!]$. We actually show that it cannot even be bisimilar to any $\mathcal{I} \| \mathcal{J}$ with $\mathcal{I} \in [\![\mathcal{S}]\!]$ and $\mathcal{J} \in [\![\mathcal{T}]\!]$. Let us assume that there exist such $\mathcal{I}$ and $\mathcal{J}$. We make the following observations:

- $\mathcal{I}$ has to contain at least one a-run with signature $\{b\}^k\{b,c\}\{b\}^\omega$. Otherwise, it would be impossible to create the $\mathcal{I}_k$ part of $\mathcal{I}_{kl}$.
- $\mathcal{J}$ has to contain at least one a-run with signature $\{c\}^l\{b,c\}\{c\}^\omega$. Otherwise, it would be impossible to create the $\mathcal{I}_l$ part of $\mathcal{I}_{kl}$.

However, these observations mean that $\mathcal{I} \| \mathcal{J}$ contains at least one a-run with signature $\emptyset^k\{c\}\emptyset^{l-k-1}\{b\}\emptyset^\omega$. It is thus not bisimilar to $\mathcal{I}_{kl}$. □

Given that we cannot have (5), the revised goal is to have a *sound* composition operator for which the right-to-left inclusion holds in (5). For NAA $\mathcal{A}_1 = (S_1, S_1^0, \mathrm{Tran}_1)$, $\mathcal{A}_2 = (S_2, S_2^0, \mathrm{Tran}_2)$, we define $\mathcal{A}_1 \| \mathcal{A}_2 = (S, S^0, \mathrm{Tran})$ with $S = S_1 \times S_2$, $S^0 = S_1^0 \times S_2^0$, and for all $(s_1, s_2) \in S$, $\mathrm{Tran}(s_1, s_2) = \{M_1 \| M_2 \mid M_1 \in \mathrm{Tran}_1(s_1), M_2 \in \mathrm{Tran}_2(s_2)\}$, where $M_1 \| M_2 = \{(a, (t_1, t_2)) \mid (a, t_1) \in M_1, (a, t_2) \in M_2\}$. Composition for DMTS is defined using the translations to and from NAA; note that this may incur an exponential blow-up.

Lemma 23. *Up to $\equiv_{\mathsf{m}}$, the operator $\|$ on NAA is associative and commutative, distributes over $\vee$, and has unit U, where U is the LTS $(\{s\}, s, \longrightarrow)$ with $s \xrightarrow{a} s$ for all $a \in \Sigma$.*

Proof. Associativity and commutativity are clear. To show distributivity over $\vee$, let $\mathcal{A}_i = (S_i, S_i^0, \mathrm{Tran}_i)$, for $i = 1, 2, 3$, be NAA. We prove that $\mathcal{A}_1 \| (\mathcal{A}_1 \vee \mathcal{A}_3) \equiv_{\mathsf{m}} \mathcal{A}_1 \| \mathcal{A}_2 \vee \mathcal{A}_1 \| \mathcal{A}_3$; right-distributivity will follow by commutativity. The state spaces of both sides are $S_1 \times S_2 \cup S_1 \times S_3$, and it is easily verified that the identity relation is a two-sided modal refinement.

For the claim that $\mathcal{A} \| \mathsf{U} \equiv_{\mathsf{m}} \mathcal{A}$ for all NAA $\mathcal{A} = (S, S^0, \mathrm{Tran})$, let u be the unique state of U and define $R = \{((s, u), s) \mid s \in S\} \subseteq S \times \mathsf{U} \times S$. We show that R is a two-sided modal refinement. Let $((s, u), s) \in R$ and $M \in \mathrm{Tran}(s, u)$, then there must be $M_1 \in \mathrm{Tran}(s)$ for which $M = M_1 \| (\Sigma \times \{u\})$. Thus $M_1 = \{(a, t) \mid (a, (t, u)) \in M\}$. Then any element of M has a corresponding one in M_1, and vice versa, and their states are related by R. For the other direction, let

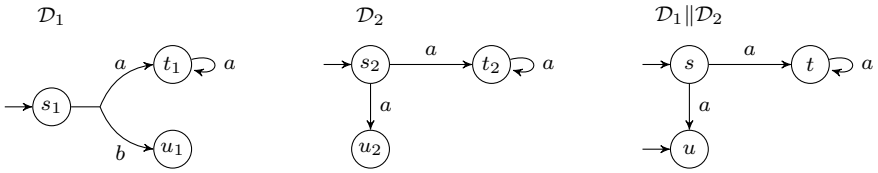

Figure 12: Two DMTS and the reachable parts of the DMTS translation of their composition. Here, $s = \{(a, (t_1, t_2)), (a, (t_1, u_2))\}$, $t = \{(a, (t_1, t_2))\}$ and $u = \emptyset$

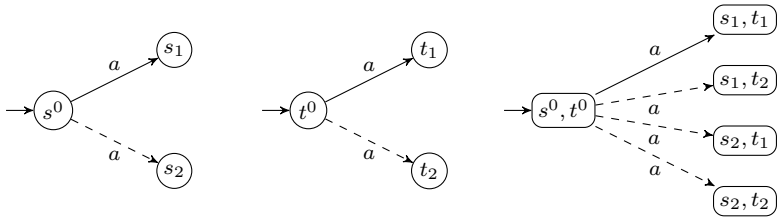

Figure 13: Two MTS and their MTS composition according to [33]

$M_1 \in \mathrm{Tran}(s)$, then $M = M_1 \| (\Sigma \times \{u\}) = \{(a, (t, u)) \mid (a, t) \in M_1\} \in \mathrm{Tran}(s, u)$, and the same argument applies. □

The next theorem is one of *independent implementability*, as it ensures that a composition of refinements is a refinement of compositions:

Theorem 24. *For all specifications $\mathcal{S}_1$, $\mathcal{S}_2$, $\mathcal{S}_3$, $\mathcal{S}_4$, $\mathcal{S}_1 \leq_{\mathsf{m}} \mathcal{S}_3$ and $\mathcal{S}_2 \leq_{\mathsf{m}} \mathcal{S}_4$ imply $\mathcal{S}_1 \| \mathcal{S}_2 \leq_{\mathsf{m}} \mathcal{S}_3 \| \mathcal{S}_4$.*

Proof. Let $\mathcal{S}_1 \leq_{\mathsf{m}} \mathcal{S}_3$ and $\mathcal{S}_2 \leq_{\mathsf{m}} \mathcal{S}_4$, then $\mathcal{S}_1 \vee \mathcal{S}_3 \equiv_{\mathsf{m}} \mathcal{S}_3$ and $\mathcal{S}_2 \vee \mathcal{S}_4 \equiv_{\mathsf{m}} \mathcal{S}_4$. By distributivity, $\mathcal{S}_3 \| \mathcal{S}_4 \equiv_{\mathsf{m}} (\mathcal{S}_1 \vee \mathcal{S}_3) \| (\mathcal{S}_2 \vee \mathcal{S}_4) \equiv_{\mathsf{m}} \mathcal{S}_1 \| \mathcal{S}_2 \vee \mathcal{S}_1 \| \mathcal{S}_3 \vee \mathcal{S}_3 \| \mathcal{S}_2 \vee \mathcal{S}_3 \| \mathcal{S}_4$, thus $\mathcal{S}_1 \| \mathcal{S}_2 \vee \mathcal{S}_1 \| \mathcal{S}_3 \vee \mathcal{S}_3 \| \mathcal{S}_2 \leq_{\mathsf{m}} \mathcal{S}_3 \| \mathcal{S}_4$. But $\mathcal{S}_1 \| \mathcal{S}_2 \leq_{\mathsf{m}} \mathcal{S}_1 \| \mathcal{S}_2 \vee \mathcal{S}_1 \| \mathcal{S}_3 \vee \mathcal{S}_3 \| \mathcal{S}_2$, finishing the argument. □

Example 3. An example of composition is shown in Fig. 12. Here the DMTS translation of $\mathcal{D}_1 \| \mathcal{D}_2$ has two initial states; it can be shown that no DMTS with a single initial state is thoroughly equivalent.

Remark that NAA composition is more precise than the composition for MTS introduced in [33]. The MTS composition is given by the following rules: $(s_1, s_2) \overset{a}{\dashrightarrow} (t_1, t_2)$ whenever $s_1 \overset{a}{\dashrightarrow} t_1$ and $s_2 \overset{a}{\dashrightarrow} t_2$, $(s_1, s_2) \xrightarrow{a} (t_1, t_2)$ whenever $s_1 \xrightarrow{a} t_1$ and $s_2 \xrightarrow{a} t_2$. The difference between the two compositions is illustrated in Fig. 13. The figure shows two MTS and their MTS composition; for their NAA composition,

$$\begin{aligned}\mathrm{Tran}(s,t) = \big\{ &\{(a,(s_1,t_1))\},\\ &\{(a,(s_1,t_1)),(a,(s_1,t_2))\},\{(a,(s_1,t_1)),(a,(s_2,t_1))\},\\ &\{(a,(s_1,t_1)),(a,(s_1,t_2)),(a,(s_2,t_1)),(a,(s_2,t_2))\}\big\}. \quad (6)\end{aligned}$$

The NAA translation of their MTS composition has eight transition constraints instead of four; note how the four constraints in (6) precisely correspond to the four implementation choices for s^0 and t^0.

It can easily be shown that generally, NAA composition is a refinement of MTS composition. The following lemma shows a stronger relationship, namely that the MTS composition is a conservative approximation of the NAA composition.

Lemma 25. *Let $\mathcal{M}_1$, $\mathcal{M}_2$, $\mathcal{M}_3$ be MTS and let $\|_{\mathrm{M}}$ and $\|_{\mathrm{A}}$ be the MTS and NAA composition, respectively. It holds that $\mathcal{M}_1\|_{\mathrm{M}}\mathcal{M}_2 \leq_{\mathsf{m}} \mathcal{M}_3$ iff $\mathcal{M}_1\|_{\mathrm{A}}\mathcal{M}_2 \leq_{\mathsf{m}} \mathcal{M}_3$.*

Proof. Let $\mathcal{M}_i = (S_i^0, \{s_i^0\}, \dashrightarrow_i, \longrightarrow_i)$ be MTS for $i = 1, 2, 3$. In the following, we use the notation $s_1\|_{\mathrm{A}}s_2$ to denote the states (s_1, s_2) of $\mathcal{M}_1\|_{\mathrm{A}}\mathcal{M}_2$ and similarly for $\|_{\mathrm{M}}$.

For a MTS translated into NAA, the Tran sets have a special structure, namely that for all states s, $\mathrm{Tran}(s)$ always has a maximal element $\{(a,t) \mid s \overset{a}{\dashrightarrow} t\}$ and a minimal element $\{(a,t) \mid s \overset{a}{\longrightarrow} t\}$ (with respect to set inclusion; *cf.* Lemma 14 for the similar property for DMTS). Furthermore, we note that $\mathrm{Tran}(s_1\|_{\mathrm{A}}s_2) \subseteq \mathrm{Tran}(s_1\|_{\mathrm{M}}s_2)$ and, moreover, $\mathrm{Tran}(s_1\|_{\mathrm{A}}s_2)$ also has a minimal and a maximal element and these elements correspond to the minimal and maximal element of $\mathrm{Tran}(s_1\|_{\mathrm{M}}s_2)$.

The fact that $\mathcal{M}_1\|_{\mathrm{A}}\mathcal{M}_2 \leq_{\mathsf{m}} \mathcal{M}_1\|_{\mathrm{M}}\mathcal{M}_2$ follows from the observation about the inclusion relation between Tran sets directly. This proves the 'only-if' part of the lemma.

To prove the 'if' part of the lemma, we let $R = \{(s_1\|_{\mathrm{M}}s_2, s_3) \mid s_1\|_{\mathrm{A}}s_2 \leq_{\mathsf{m}} s_3\}$ and show that it is a modal refinement relation witnessing $\mathcal{M}_1\|_{\mathrm{M}}\mathcal{M}_2 \leq_{\mathsf{m}} \mathcal{M}_3$. Let $(s_1\|_{\mathrm{M}}s_2, s_3) \in R$.

- Let $s_1\|_{\mathrm{M}}s_2 \overset{a}{\dashrightarrow} t_1\|_{\mathrm{M}}t_2$. Then $(a, t_1\|_{\mathrm{M}}t_2)$ belongs to the maximal element of $\mathrm{Tran}(s_1\|_{\mathrm{M}}s_2)$, which is also in $\mathrm{Tran}(s_1\|_{\mathrm{A}}s_2)$. Due to $s_1\|_{\mathrm{A}}s_2 \leq_{\mathsf{m}} s_3$ we have some $N \in \mathrm{Tran}(s_3)$ with $(a, t_3) \in N$ such that $t_1\|_{\mathrm{A}}t_2 \leq_{\mathsf{m}} t_3$. Thus $s_3 \overset{a}{\dashrightarrow} t_3$ and $(t_1\|_{\mathrm{M}}t_2, t_3) \in R$.

- Let $s_3 \overset{a}{\longrightarrow} t_3$. Then all elements of $\mathrm{Tran}(s_3)$ contain (a, t_3). If we now chose the minimal element $M \in \mathrm{Tran}(s_1\|_{\mathrm{A}}s_2)$ then it has to contain $(a, t_1\|_{\mathrm{A}}t_2)$ such that $(t_1\|_{\mathrm{A}}t_2 \leq_{\mathsf{m}} t_3)$. This means that $s_1\|_{\mathrm{M}}s_2 \overset{a}{\longrightarrow} t_1$ and $(t_1\|_{\mathrm{M}}t_2, t_3) \in R$. □

4.3. Quotient

The quotient operator for a specification theory is used to synthesise specifications for components of a composition. Hence it is to have the property, for all specifications $\mathcal{S}$, $\mathcal{S}_1$ and all implementations $\mathcal{I}_1$, $\mathcal{I}_2$, that

$$\mathcal{I}_1 \in [\![\mathcal{S}_1]\!] \text{ and } \mathcal{I}_2 \in [\![\mathcal{S}/\mathcal{S}_1]\!] \text{ imply } \mathcal{I}_1\|\mathcal{I}_2 \in [\![\mathcal{S}]\!]. \tag{7}$$

Furthermore, $\mathcal{S}/\mathcal{S}_1$ is to be as permissive as possible.

4.3.1. Quotient for MTS

Before we describe the general construction of the quotient, we start with a simpler construction that works for the important special case of MTS. However, MTS are not closed under quotient, *cf.* [34, Thm. 5.5]; we show that the quotient of two MTS will generally be a DMTS.

Recall that MTS have only one initial state and all their must transitions are singletons. Let $\mathcal{M}_1 = (S_1, s_1^0, \dashrightarrow_1, \longrightarrow_1)$ and $\mathcal{M}_2 = (S_2, s_2^0, \dashrightarrow_2, \longrightarrow_2)$ be

MTS. We define $\mathcal{M}_1/\mathcal{M}_2 = (S, s^0, \dashrightarrow, \longrightarrow)$ with $S = 2^{S_1 \times S_2}$, $s^0 = \{(s_1^0, s_2^0)\}$, and the transition relations given as follows.

We first define $\emptyset \overset{a}{\dashrightarrow} \emptyset$ for all $a \in \Sigma$. There are no must transitions from $\emptyset$. For $s = \{(s_1^1, s_2^1), \ldots, (s_1^n, s_2^n)\} \in S$ we say that $a \in \Sigma$ is *permissible from* s if for all $i = 1, \ldots, n$ either $s_1^i \overset{a}{\dashrightarrow}$ or $s_2^i \overset{a}{\not\dashrightarrow}$.

For a permissible from s and $i \in \{1, \ldots, n\}$, let $\{t_2^{i,1}, \ldots, t_2^{i,m_i}\} = \{t_2 \in S_2 \mid s_2 \overset{a}{\dashrightarrow} t_2\}$ be an enumeration of the possible states in S_2 after an a-transition. We then define $pt_a(s) = \left\{\{(t_1^{i,j}, t_2^{i,j}) \mid i = 1, \ldots, n, j = 1, \ldots, m_i\} \;\middle|\; \forall i : \forall j : s_1^i \overset{a}{\dashrightarrow} t_1^{i,j})\right\}$.

The transitions of s are now given as follows: for every a permissible from s and every $t \in pt_a(s)$, let $s \overset{a}{\dashrightarrow} t$. Furthermore, for every $s_1^i \overset{a}{\longrightarrow} t_1$ let $s \longrightarrow \{(a, M) \in \{a\} \times pt_a(s) \mid \exists t_2 : (t_1, t_2) \in M, s_2^i \overset{a}{\longrightarrow} t_2\}$.

Recall from Lemma 25 that the MTS composition is a conservative approximation to the NAA compositions. This means that the following theorem holds regardless of which of the two compositions is used.

Theorem 26. *For all MTS specifications $\mathcal{M}_1$, $\mathcal{M}_2$ and $\mathcal{M}_3$, $\mathcal{M}_1 \| \mathcal{M}_2 \leq_{\mathsf{m}} \mathcal{M}_3$ iff $\mathcal{M}_2 \leq_{\mathsf{m}} \mathcal{M}_3/\mathcal{M}_1$.*

Proof. In this proof only, let $\|$ denote MTS composition.

Write $\mathcal{M}_i = (S_i, s_i^0, \dashrightarrow_i, \longrightarrow_i)$ for $i = 1, 2, 3$. We use the following notation to help distinguish states of $\mathcal{M}_1 \| \mathcal{M}_2$ and $\mathcal{M}_3/\mathcal{M}_1$. The states of $\mathcal{M}_1 \| \mathcal{M}_2$ are denoted by $s_1 \| s_2$ instead of (s_1, s_2) while the states of $\mathcal{M}_3/\mathcal{M}_1$ are denoted by $\{s_3/s_1, \ldots\}$ instead of $\{(s_3, s_1), \ldots\}$. We also note that for states of $\mathcal{M}_3/\mathcal{M}_1$, $s \supseteq t$ implies $s \leq_{\mathsf{m}} t$ due to the construction.

Now assume that $\mathcal{M}_2 \leq_{\mathsf{m}} \mathcal{M}_3/\mathcal{M}_1$ and let $R = \{(s_1 \| s_2, s_3) \mid s_2 \leq_{\mathsf{m}} \{s_3/s_1\}\}$. We show that R is a witness for $\mathcal{M}_1 \| \mathcal{M}_2 \leq_{\mathsf{m}} \mathcal{M}_3$, *i.e.* that it satisfies the conditions of Definition 2. Let $(s_1 \| s_2, s_3) \in R$.

- Let $s_1 \| s_2 \overset{a}{\dashrightarrow} t_1 \| t_2$. As $s_2 \leq_{\mathsf{m}} \{s_3/s_1\}$ this means that $\{s_3/s_1\} \overset{a}{\dashrightarrow} \{t_3^1/t_1^1, \ldots, t_3^k/t_1^k\} = t$ and $t_2 \leq_{\mathsf{m}} t$. Due to the construction of $\{s_3/s_1\}$, we know that there is an index j for which $t_1^j = t_1$ and $s_3 \overset{a}{\dashrightarrow} t_3^j$. Let $t_3 = t_3^j$. As $t \supseteq \{t_3/t_1\}$, $x' \leq_{\mathsf{m}} \{s'/t'\}$. Therefore, $(t_1 \parallel t_2, t_3) \in R$.
- Let $s_3 \overset{a}{\longrightarrow} t_3$. This means that $\{s_3/s_1\} \longrightarrow U$. As $s_2 \leq_{\mathsf{m}} \{s_3/s_1\}$, we know that $s_2 \overset{a}{\longrightarrow} t_2$ and $t_2 \leq_{\mathsf{m}} u$ for some $(a, u) \in U$. Due to the construction of U we know that there exists $t_3/t_1 \in u$. Again, as $u \supseteq \{t_3/t_1\}$, $t_2 \leq_{\mathsf{m}} \{t_3/t_1\}$. Therefore, $(t_1 \parallel t_2, t_3) \in R$.

Assume, for the other direction of the proof, that $\mathcal{M}_1 \parallel \mathcal{M}_2 \leq_{\mathsf{m}} \mathcal{M}_3$. Define

$$R = \{(s_2, \{s_3^1/s_1^1, \ldots, s_3^n/s_1^n\}) \mid \forall i = 1, \ldots, n : s_1^i \parallel s_2 \leq_{\mathsf{m}} s_3^i\};$$

note that $(s_2, \emptyset) \in R$ for all $s_2 \in S_2$. We show that R is a witness for $\mathcal{M}_2 \leq_{\mathsf{m}} \mathcal{M}_3/\mathcal{M}_1$. Let $(s_2, s) \in R$ with $s = \{s_3^1/s_1^1, \ldots, s_3^n/s_1^n\}$.

- Let $s_2 \overset{a}{\dashrightarrow} t_2$. If there is no i such that $s_1^i \overset{a}{\dashrightarrow}$ then $s \overset{a}{\dashrightarrow} \emptyset$ and $(t_2, \emptyset) \in R$. Otherwise, take an arbitrary $s_1^i \overset{a}{\dashrightarrow} t_1^{i,j}$. We have $s_1^i \| s_2 \overset{a}{\dashrightarrow} t_1^{i,j} \| t_2$ and as $s_1^i \| s_2 \leq_{\mathsf{m}} s_3^i$ we also have a corresponding $s_3^i \overset{a}{\dashrightarrow} t_3^{i,j}$ with $t_1^{i,j} \| t_2 \leq_{\mathsf{m}} t_3^{i,j}$. We fix these $t_3^{i,j}$. Let $t = \{t_3^{i,j}/t_1^{i,j} \mid i \in \{1, \ldots, n\}, j \in \{1, \ldots, m_i\}\}$. Clearly, $s \overset{a}{\dashrightarrow} t$ and $(t_2, t) \in R$.

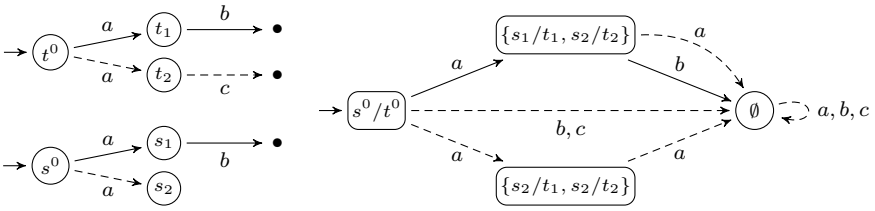

Figure 14: Two nondeterministic MTS and their quotient

- Let $s \longrightarrow U$ and let $s_3^i \xrightarrow{a} t_3^i$ be the corresponding must transition in the construction. As $s_1^i \| s_2 \leq_{\mathsf{m}} s_3^i$, this means that $s_2 \xrightarrow{a} t_2$ and $s_1^i \xrightarrow{a} t_1^i$ such that $t_1^i \| t_2 \leq_{\mathsf{m}} t_3^i$. This also means that $s_2 \overset{a}{\dashrightarrow} t_2$. We thus build t as we did in the previous case. Let $\bar{t} = \{\bar{t}_3/\bar{t}_1 \in t \mid \bar{t}_1 \neq t_1^i\} \cup \{t_3^i/t_1^i\}$. Due to the construction of must transitions, $\bar{t} \in U$. Clearly $(t_2, \bar{t}) \in R$. □

Example 4. We illustrate the construction on an example. Let S and T be the MTS in the left part of Fig. 14. We construct S/T; the end result is displayed in the right part of the figure.

First we construct the may-successors of s^0/t^0. Under b and c there are no constraints, hence we go to $\emptyset$. For a, we have all permutations of assignments of successors of s to successors of t, namely $\{s_1/t_1, s_1/t_2\}$, $\{s_1/t_1, s_2/t_2\}$, $\{s_2/t_1, s_1/t_2\}$ and $\{s_2/t_1, s_2/t_2\}$. Since there is a must-transition from s^0 (to s_1), we create a disjunctive must-transition to all successors that can be used to yield a must-transition when composed with the must-transition from t^0 to t_1. These are all successors where t_1 is mapped to s_1, hence the first two. However, $\{s_1/t_1, s_1/t_2\}$ will turn out inconsistent, as it requires to refine s_1 by a composition with t_2. As t_2 has no must under b, the composition has none either, hence the must of s_1 can never be matched. As a result, after pruning, the disjunctive must from $\{s^0/t^0\}$ leads only to $\{s_1/t_1, s_2/t_2\}$. Further, $\{s_2/t_1, s_1/t_2\}$ is inconsistent for the same reason, so that we only have one other may-transition under a from $\{s^0/t^0\}$.

Now $\{s_1/t_1, s_2/t_2\}$ is obliged to have a must under b so that it refines s_1 when composed with t_1, but cannot have any c in order to match s_2 when composed with t_2. Similarly, $\{s_2/t_1, s_2/t_2\}$ has neither c nor b. One can easily verify that $T \| (S/T) \equiv_{\mathsf{m}} S$ in this case. □

4.3.2. Quotient for NAA

We now introduce the general quotient operator for NAA. The construction is similar to the previous one, with the notions of permissibility and $pt_a(s)$ adapted to the more general setting.

Let $\mathcal{A}_1 = (S_1, S_1^0, \mathrm{Tran}_1)$, $\mathcal{A}_2 = (S_2, S_2^0, \mathrm{Tran}_2)$ be NAA and define $\mathcal{A}_1/\mathcal{A}_2 = (S, \{s^0\}, \mathrm{Tran})$, with $S = 2^{S_1 \times S_2}$, $s^0 = \{(s_1^0, s_2^0) \mid s_1^0 \in S_1^0, s_2^0 \in S_2^0\}$, and Tran given as follows:

Let $\mathrm{Tran}(\emptyset) = 2^{\Sigma \times \{\emptyset\}}$. For $s = \{(s_1^1, s_2^1), \ldots, (s_1^n, s_2^n)\} \in S$, say that $a \in \Sigma$ is *permissible from* s if it holds for all $i = 1, \ldots, n$ that there is $M_1 \in \mathrm{Tran}_1(s_1^i)$ and $t_1 \in S_1$ for which $(a, t_1) \in M_1$, or else there is no $M_2 \in \mathrm{Tran}_2(s_2^i)$ and no $t_2 \in S_2$ for which $(a, t_2) \in M_2$.

For a permissible from s and $i \in \{1,\ldots,n\}$, let $\{t_2^{i,1},\ldots,t_2^{i,m_i}\} = \{t_2 \in S_2 \mid \exists M_2 \in \mathrm{Tran}_2(s_2^i) : (a,t_2) \in M_2\}$ be an enumeration of the possible states in S_2 after an a-transition and define $pt_a(s) = \{\{(t_1^{i,j}, t_2^{i,j}) \mid i = 1,\ldots,n, j = 1,\ldots,m_i\} \mid \forall i : \forall j : \exists M_1 \in \mathrm{Tran}_1(s_1^i) : (a, t_1^{i,j}) \in M_1\}$, the set of all sets of possible assignments of next-a states from s_1^i to next-a states from s_2^i.

Now let $pt(s) = \{(a,t) \mid t \in pt_a(s), a \text{ admissible from } s\}$ and define $\mathrm{Tran}(s) = \{M \subseteq pt(s) \mid \forall i = 1,\ldots,n : \forall M_2 \in \mathrm{Tran}_2(s_2^i) : M \triangleright M_2 \in \mathrm{Tran}_1(s_1^i)\}$. Here $\triangleright$ is the composition-projection operator defined by $M \triangleright M_2 = \{(a, t \triangleright t_2) \mid (a,t) \in M, (a,t_2) \in M_2\}$ and $t \triangleright t_2 = \{(t_1^1, t_2^1), \ldots, (t_1^k, t_2^k)\} \triangleright t_2^i = t_1^i$ (note that by construction, there is precisely one pair in t whose second component is t_2^i).

Theorem 27. *For all specifications $\mathcal{S}_1$, $\mathcal{S}_2$, $\mathcal{S}_3$, $\mathcal{S}_1 \| \mathcal{S}_2 \leq_{\mathsf{m}} \mathcal{S}_3$ iff $\mathcal{S}_2 \leq_{\mathsf{m}} \mathcal{S}_3/\mathcal{S}_1$.*

Proof. We show the proof for NAA. Let $\mathcal{A}_1 = (S_1, S_1^0, \mathrm{Tran}_1)$ and $\mathcal{A}_2 = (S_2, S_2^0, \mathrm{Tran}_2)$, $\mathcal{A}_3 = (S_3, S_3^0, \mathrm{Tran}_3)$; we show that $\mathcal{A}_1 \| \mathcal{A}_2 \leq_{\mathsf{m}} \mathcal{A}_3$ iff $\mathcal{A}_2 \leq_{\mathsf{m}} \mathcal{A}_3/\mathcal{A}_1$.

We assume that the elements of $\mathrm{Tran}_1(s_1)$ are pairwise disjoint for each $s_1 \in S_1$; this can be achieved by, if necessary, splitting states.

We also use the notation introduced in the proof of Theorem 26, *i.e.* $s_1 \| s_2$ instead of (s_1, s_2) when speaking about states of $\mathcal{A}_1 \| \mathcal{A}_2$ and $\{s_3/s_1, \ldots\}$ instead of $\{(s_3, s_1), \ldots\}$ when speaking about states of $\mathcal{A}_3/\mathcal{A}_1$. We further note that by construction, $s \supseteq t$ implies $s \leq_{\mathsf{m}} t$ for all $s, t \in 2^{S_3 \times S_1}$.

Now assume that $\mathcal{A}_2 \leq_{\mathsf{m}} \mathcal{A}_3/\mathcal{A}_1$ and let $R = \{(s_1 \| s_2, s_3) \mid s_2 \leq_{\mathsf{m}} \{s_3/s_1\}\}$; we show that R is a witness for $\mathcal{A}_1 \| \mathcal{A}_2 \leq_{\mathsf{m}} \mathcal{A}_3$.

Let $(s_1 \| s_2, s_3) \in R$ and $M_\| \in \mathrm{Tran}_\|(s_1 \| s_2)$. Then $M_\| = M_1 \| M_2$ with $M_1 \in \mathrm{Tran}_1(s_1)$ and $M_2 \in \mathrm{Tran}_2(s_2)$. As $s_2 \leq_{\mathsf{m}} \{s_3/s_1\}$, we can pair M_2 with an $M_/ \in \mathrm{Tran}_/(\{s_3/s_1\})$, *i.e.* such that the conditions in (2) are satisfied (see Definition 7).

Let $M_3 = M_/ \triangleright M_1$. We show that (2) holds for the pair $M_\|, M_3$:

- Let $(a, t_1 \| t_2) \in M_\|$, then there are $(a,t_1) \in M_1$ and $(a,t_2) \in M_2$. By (2), there is $(a,t) \in M_/$ such that $t_2 \leq_{\mathsf{m}} t$. Write $t = \{t_3^1/t_1^1, \ldots, t_3^n/t_1^n\}$. By construction, there is an index i for which $t_1^i = t_1$, hence $(a, t_3^i) \in M_3$. Also, $t \supseteq \{t_3^i/t_1^i\}$, hence $t_2 \leq_{\mathsf{m}} \{t_3^i/t_1^i\}$ and consequently $(t_1 \| t_2, t_3) \in R$.
- Let $(a, t_3) \in M_3$, then there are $(a,t) \in M_/$ and $(a,t_1) \in M_1$ such that $t_3/t_1 \in t$. By (2), there is $(a,t_2) \in M_2$ for which $t_2 \leq_{\mathsf{m}} t$. Thus $(a, t_1 \| t_2) \in M$, and by $t \supseteq \{t_3/t_1\}$, $t_2 \leq_{\mathsf{m}} \{t_3/t_1\}$.

Assume, for the other direction of the proof, that $\mathcal{A}_1 \| \mathcal{A}_2 \leq_{\mathsf{m}} \mathcal{A}_3$. Define $R \subseteq S_2 \times 2^{S_3 \times S_1}$ by

$$R = \{(s_2, \{s_3^1/s_1^1, \ldots, s_3^n/s_1^n\}) \mid \forall i = 1,\ldots,n : s_1^i \| s_2 \leq_{\mathsf{m}} s_3^i\};$$

we show that R is a witness for $\mathcal{A}_2 \leq_{\mathsf{m}} \mathcal{A}_3/\mathcal{A}_1$. Let $(s_2, s) \in R$, with $s = \{s_3^1/s_1^1, \ldots, s_3^n/s_1^n\}$, and $M_2 \in \mathrm{Tran}_2(s_2)$.

For every $i = 1,\ldots,n$, write $\mathrm{Tran}_1(s_1^i) = \{M_1^{i,1}, \ldots, M_1^{i,m_i}\}$. By assumption, $M_1^{i,j_1} \cap M_1^{i,j_2} = \emptyset$ for $j_1 \neq j_2$, hence every $(a,t_1) \in\in \mathrm{Tran}_1(s_1^i)$ is contained in a unique $M_1^{i,\delta_i(a,t_1)} \in \mathrm{Tran}_1(s_1^i)$.

For every $j = 1,\ldots,m_i$, let $M^{i,j} = M_1^{i,j} \| M_2 \in \mathrm{Tran}_\|(s_1^i \| s_2)$. By $s_1^i \| s_2 \leq_{\mathsf{m}} s_3^i$, we have $M_3^{i,j} \in \mathrm{Tran}_3(s_3^i)$ such that (2) holds for the pair $M^{i,j}, M_3^{i,j}$.

Now define

$$M = \{(a,t) \mid \exists (a,t_2) \in M_2 : \forall t_3/t_1 \in t : \exists i : \exists M_1 \in \mathrm{Tran}_1(s_1^i) : \\ (a,t_1) \in M_1, (a,t_3) \in M_3^{i,\delta_i(a,t_1)}, t_1 \| t_2 \leq_{\mathsf{m}} t_3\}\,. \quad (8)$$

We need to show that $M \in \mathrm{Tran}_/(s)$.

Let $i \in \{1,\ldots,n\}$ and $M_1^{i,j} \in \mathrm{Tran}_1(s_1^i)$; we claim that $M \triangleright M_1^{i,j}$ and $M_3^{i,j}$ are related as in (2). Let $(a,t_3) \in M \triangleright M_1^{i,j}$, then $t_3/t_1 \in t$, $(a,t_1) \in M_1^{i,j}$ and $(a,t) \in M$. By disjointness, $j = \delta_i(a,t_1)$, hence by definition of M, $(a,t_3) \in M_3^{i,j}$ as was to be shown.

For the reverse inclusion, let $(a,t_3) \in M_3^{i,j}$. By (2) and definition of $M^{i,j}$, there are $(a,t_1) \in M_1^{i,j}$ and $(a,t_2) \in M_2$ for which $t_1 \| t_2 \leq_{\mathsf{m}} t_3$. Thus $j = \delta_i(a,t_1)$, so that there must be $(a,t) \in M$ for which $t_3/t_1 \in t$, but then also $(a,t_3) \in M \triangleright M_1^{i,j}$.

We show that M_2 and M are related as in (2).

- Let $(a,t_2) \in M_2$. For every $i = 1,\ldots,n$, $M_1 \in \mathrm{Tran}_1(t_1^i)$ and $(a,t_1) \in M$, we can use (2) to choose an element $(a,\tau_i(a,t_1)) \in M_3^{i,\delta_i(a,t_1)}$ for which $t_1 \| t_2 \leq_{\mathsf{m}} \tau_i(a,t_1)$. Let $t = \{\tau_i(a,t_1)/t_1 \mid i = 1,\ldots,n, \exists M_1 \in \mathrm{Tran}_1(t_1^i) : (a,t_1) \in M_1\}$, then $(a,t) \in M$ and $(t_2,t) \in R$.
- Let $(a,t) \in M$, then we have $(a,t_2) \in M_2$ satisfying the conditions in (8). Hence $t_1 \| t_2 \leq_{\mathsf{m}} t_3$ for all $t_3/t_1 \in t$, so that $(t_2,t) \in R$. □

As a corollary, we get (7): If $\mathcal{I}_2 \in [\![\mathcal{S}/\mathcal{S}_1]\!]$, *i.e.* $\mathcal{I}_2 \leq_{\mathsf{m}} \mathcal{S}/\mathcal{S}_1$, then $\mathcal{S}_1 \| \mathcal{I}_2 \leq_{\mathsf{m}} \mathcal{S}$, which using $\mathcal{I}_1 \leq_{\mathsf{m}} \mathcal{S}_1$ and Theorem 24 implies $\mathcal{I}_1 \| \mathcal{I}_2 \leq_{\mathsf{m}} \mathcal{S}_1 \| \mathcal{I}_2 \leq_{\mathsf{m}} \mathcal{S}$. The reverse implication in Theorem 27 implies that $\mathcal{S}/\mathcal{S}_1$ is as permissive as possible.

Corollary 28. *With operations $\wedge$, $\vee$, $\|$ and $/$, each of our four classes of specifications forms a commutative residuated lattice up to $\equiv_{\mathsf{m}}$.*

Proof. We have already seen in Corollary 21 that the class of NAA forms a lattice, up to $\equiv_{\mathsf{m}}$, under $\wedge$ and $\vee$, and by Theorem 27, $/$ is the residual, up to $\equiv_{\mathsf{m}}$, of $\|$. All other properties (such as distributivity of $\|$ over $\vee$ or $\mathcal{N} \| \bot \equiv_{\mathsf{m}} \bot$) follow. □

5. Related Work

The modal ν-calculus is equivalent to the Hennessy-Milner logic with *greatest* fixed points, which arises from Hennessy-Milner logic (HML) [28] by introducing variables and greatest fixed points. If also *least* fixed points are allowed, one arrives at the full *modal μ-calculus* [30, 41, 49]. Janin and Walukiewicz have in [29] introduced an automata-like representation for the modal μ-calculus which seems related to our NAA.

DMTS have been proposed as solutions to algebraic process equations in Larsen and Xinxin's [37] and further investigated also as a specification formalism [4, 34]. The DMTS formalism is a member of the modal transition systems (MTS) family and as such has also received attention recently. The MTS formalisms have proven to be useful in practice. Industrial applications started with Bruns' [12] where MTS have been used for an air-traffic system at

Heathrow airport. Besides, MTS classes are advocated as an appropriate base for interface theories by Raclet *et al.* in [46] and for product line theories in Nyman's [39]. Further, an MTS based software engineering methodology for design via merging partial descriptions of behaviour has been established by Uchitel and Chechik in [50] and methods for supervisory control of MTS shown by Darondeau *et al.* in [18]. Tool support is quite extensive, *e.g.* [3, 10, 19, 32].

Over the years, many extensions of MTS have been proposed. While MTS can only specify whether or not a particular transition is required, some extensions equip MTS with more general abilities to describe what *combinations* of transitions are possible. These include DMTS [37], Fecher and Schmidt's 1-MTS [21] allowing to express exclusive disjunction, OTS [8] capable of expressing positive Boolean combinations, and Boolean MTS [6] covering all Boolean combinations. The last one is closely related to our NAA as well as hybrid modal logic [9, 42]. Our results show that all these formalisms are at most as expressive as DMTS.

Larsen has shown in [33] that any finite acyclic MTS is equivalent to a HML formula (without recursion or fixed points), the *characteristic formula* of the given MTS, *cf.* (3). Conversely, Boudol and Larsen show in [11] that any consistent and *prime* HML formula is equivalent to a MTS.[4] Here we extend these results to ν-calculus formulae, and show that any such formula is equivalent to a DMTS, solving a problem left open in [37]. Hence the modal ν-calculus supports full compositionality and decomposition in the sense of [34]. This finishes some of the work started in [11, 33, 34]. Recently, the graphical representability of a variant of alternating simulation called covariant-contravariant simulation has been studied in [1].

Quotients are related to *decomposition* of processes and properties, an issue which has received considerable attention through the years. In [37], a solution to bisimulation $C(X) \sim P$ for a given process P and context C is provided (as a DMTS). This solves the quotienting problem P/C for the special case where both P and C are processes. This is extended in [36] to the setting where the context C can have several holes and $C(X_1, \ldots, X_n)$ must satisfy a ν-calculus property Q. However, C remains to be a process context, not a specification context. Our *specification* context allows for arbitrary specifications, representing infinite sets of processes and process equations. Other extensions use infinite conjunctions [23], probabilistic processes [24] or processes with continuous time and space [16].

Quotient operators, or *guarantee* or *multiplicative implication* as they are called there, are also well-known from various logical formalisms. Indeed, the algebraic properties of our parallel composition $\|$ and quotient / resemble closely those of multiplicative conjunction & and implication $\multimap$ in *linear logic* [25], and of spatial conjunction and implication in *spatial logic* [15] and *separation logic* [40, 47]. For these and other logics, proof systems have been developed which allow one to reason about expressions containing these operators. In these logics, & and $\multimap$ are first-class operators on par with the other logical operators, and their semantics are defined as certain sets of processes. In contrast, for NAA and hence, via the translations, also for ν-calculus, $\|$ and / are *derived* operators, and we provide constructions to reduce any expression which contains them, to one which does not. This is important from the perspective of reuse of

[4] A HML formula is *prime* if implying a disjunction means implying one of the alternatives.

components and useful in industrial applications. To the best of our knowledge, there are no other such *reductions* of quotient for the synchronisation type of composition in the context of specifications.

6. Conclusion

In this paper we have introduced a general specification framework whose basis consists of four different but equally expressive formalisms: one of a graphical behavioural kind (DMTS), one logic-based (ν-calculus) and two intermediate languages between the former two (NAA and hybrid modal logic). We have shown their structural equivalence.

The established connection implies several consequences. On the one hand, it allows for a graphical representation of ν-calculus. Further, composition on DMTS can be transferred to the modal ν-calculus, hence turning it into a modal process algebra. On the other hand, such a correspondence identifies a class of modal transition systems with a natural expressive power and provides another justification of this formalism. Further, this class is closed under both conjunction and disjunction, a requirement raised by component-based design methods. However, it is not closed under complement and difference.[5] Nevertheless, since DMTS are closed under conjunction, disjunction and composition, we still have a positive Boolean process algebra.

Altogether, we have shown that the framework possesses a rich algebraic structure that includes logical (conjunction, disjunction) and behavioural operations (parallel composition and quotient) and forms a complete specification theory in the sense of [2, 34].

Moreover, the construction of the quotient solves an open problem in the area of MTS. All attempts to find the quotient for variants of MTS so far have been limited to the much simpler deterministic case [45]. Here we have given the first solution to the quotient on nondeterministic MTS: firstly a solution to the general case of nondeterministic DMTS and, secondly, an exponentially better algorithm for nondeterministic MTS. Due to the established correspondence, the quotient can be applied also to ν-calculus formulae. We remark that all our translations and constructions are based on a new *normal form* for ν-calculus expressions, and that turning a ν-calculus expression into normal form may incur an exponential blow-up. However, the translations and constructions preserve the normal form, so that this translation only need be applied once in the beginning.

As for future work, we hope to establish the exact complexity of the quotient constructions. We conjecture that the exponential blow-up of the construction is in general unavoidable. Further, we plan to implement the operations detailed here within the graphical tool MoTraS [32].

Acknowledgement. The authors are indebted to several anonymous reviewers who provided very useful comments and suggestions, in particular regarding related work.

[5] Previous results on difference [48] are incorrect due to a mistake in [22] on conjunction of MTS, see [31, p. 36].

References

[1] Luca Aceto, Ignacio Fábregas, David de Frutos-Escrig, Anna Ingólfsdóttir, and Miguel Palomino. On the specification of modal systems: A comparison of three frameworks. *Sci. Comput. Program.*, 78(12):2468–2487, 2013.

[2] Sebastian S. Bauer, Alexandre David, Rolf Hennicker, Kim G. Larsen, Axel Legay, Ulrik Nyman, and Andrzej Wąsowski. Moving from specifications to contracts in component-based design. In Juan de Lara and Andrea Zisman, editors, *FASE*, volume 7212 of *Lect. Notes Comput. Sci.*, pages 43–58. Springer-Verlag, 2012.

[3] Sebastian S. Bauer, Philip Mayer, and Axel Legay. MIO workbench: A tool for compositional design with modal input/output interfaces. In Bultan and Hsiung [14], pages 418–421.

[4] Nikola Beneš, Ivana Černá, and Jan Křetínský. Modal transition systems: Composition and LTL model checking. In Bultan and Hsiung [14], pages 228–242.

[5] Nikola Beneš, Benoît Delahaye, Uli Fahrenberg, Jan Křetínský, and Axel Legay. Hennessy-Milner logic with greatest fixed points as a complete behavioural specification theory. In Pedro R. D'Argenio and Hernán C. Melgratti, editors, *CONCUR*, volume 8052 of *Lect. Notes Comput. Sci.*, pages 76–90. Springer-Verlag, 2013.

[6] Nikola Beneš, Jan Křetínský, Kim G. Larsen, Mikael H. Møller, and Jiří Srba. Parametric modal transition systems. In Bultan and Hsiung [14], pages 275–289.

[7] Nikola Beneš, Jan Křetínský, Kim G. Larsen, and Jiří Srba. On determinism in modal transition systems. *Theor. Comput. Sci.*, 410(41):4026–4043, 2009.

[8] Nikola Beneš and Jan Křetínský. Process algebra for modal transition systemses. In Ludek Matyska, Michal Kozubek, Tomáš Vojnar, Pavel Zemčík, and David Antos, editors, *MEMICS*, volume 16 of *OASICS*, pages 9–18. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Germany, 2010.

[9] Patrick Blackburn. Representation, reasoning, and relational structures: a hybrid logic manifesto. *Log. J. IGPL*, 8(3):339–365, 2000.

[10] Anders Børjesson, Kim G. Larsen, and Arne Skou. Generality in design and compositional verification using TAV. *Formal Meth. Syst. Design*, 6(3):239–258, 1995.

[11] Gérard Boudol and Kim G. Larsen. Graphical versus logical specifications. *Theor. Comput. Sci.*, 106(1):3–20, 1992.

[12] Glenn Bruns. An industrial application of modal process logic. *Sci. Comput. Program.*, 29(1-2):3–22, 1997.

[13] Glenn Bruns and Patrice Godefroid. Model checking partial state spaces with 3-valued temporal logics. In Nicolas Halbwachs and Doron Peled, editors, *CAV*, volume 1633 of *Lect. Notes Comput. Sci.*, pages 274–287. Springer-Verlag, 1999.

[14] Tevfik Bultan and Pao-Ann Hsiung, editors. *Automated Technology for Verification and Analysis, 9th Int. Symp., ATVA 2011*, volume 6996 of *Lect. Notes Comput. Sci.* Springer-Verlag, 2011.

[15] Luís Caires and Luca Cardelli. A spatial logic for concurrency. *Inf. Comp.*, 186(2), 2003.

[16] Luca Cardelli, Kim G. Larsen, and Radu Mardare. Modular Markovian logic. In Luca Aceto, Monika Henzinger, and Jiří Sgall, editors, *ICALP (2)*, volume 6756 of *Lect. Notes Comput. Sci.*, pages 380–391. Springer-Verlag, 2011.

[17] Edmund M. Clarke and E. Allen Emerson. Design and synthesis of synchronization skeletons using branching-time temporal logic. In Dexter Kozen, editor, *Logic of Programs*, volume 131 of *Lect. Notes Comput. Sci.*, pages 52–71. Springer-Verlag, 1981.

[18] Philippe Darondeau, Jérémy Dubreil, and Hervé Marchand. Supervisory control for modal specifications of services. In *WODES*, pages 428–435, 2010.

[19] Nicolás D'Ippolito, Dario Fischbein, Howard Foster, and Sebastián Uchitel. MTSA: Eclipse support for modal transition systems construction, analysis and elaboration. In L. Cheng, A. Orso, and M. P. Robillard, editors, *ETX*, pages 6–10. ACM, 2007.

[20] Uli Fahrenberg, Axel Legay, and Louis-Marie Traonouez. Structural refinement for the modal nu-calculus. In Gabriel Ciobanu and Dominique Méry, editors, *ICTAC*, volume 8687 of *Lect. Notes Comput. Sci.*, pages 169–187. Springer-Verlag, 2014.

[21] Harald Fecher and Heiko Schmidt. Comparing disjunctive modal transition systems with an one-selecting variant. *J. Logic Algebr. Program.*, 77(1-2):20–39, 2008.

[22] Dario Fischbein and Sebastián Uchitel. On correct and complete strong merging of partial behaviour models. In Mary Jean Harrold and Gail C. Murphy, editors, *SIGSOFT FSE*, pages 297–307. ACM, 2008.

[23] Wan Fokkink, Rob J. van Glabbeek, and Paulien de Wind. Compositionality of Hennessy-Milner logic by structural operational semantics. *Theor. Comput. Sci.*, 354(3):421–440, 2006.

[24] Daniel Gebler and Wan Fokkink. Compositionality of probabilistic Hennessy-Milner logic through structural operational semantics. In Maciej Koutny and Irek Ulidowski, editors, *CONCUR*, volume 7454 of *Lect. Notes Comput. Sci.*, pages 395–409. Springer-Verlag, 2012.

[25] Jean-Yves Girard. Linear logic. *Theor. Comput. Sci.*, 50:1–102, 1987.

[26] James B. Hart, Lori Rafter, and Constantine Tsinakis. The structure of commutative residuated lattices. *Internat. J. Algebra Comput.*, 12(4):509–524, 2002.

[27] Matthew Hennessy. Acceptance trees. *J. ACM*, 32(4):896–928, 1985.

[28] Matthew Hennessy and Robin Milner. Algebraic laws for nondeterminism and concurrency. *J. ACM*, 32(1):137–161, 1985.

[29] David Janin and Igor Walukiewicz. Automata for the modal mu-calculus and related results. In Jirí Wiedermann and Petr Hájek, editors, *MFCS*, volume 969 of *Lect. Notes Comput. Sci.*, pages 552–562. Springer-Verlag, 1995.

[30] Dexter Kozen. Results on the propositional μ-calculus. *Theor. Comput. Sci.*, 27, 1983.

[31] Jan Křetínský. *Modal Transition Systems: Extensions and Analysis*. PhD thesis, Masaryk University, Brno, Dept. of Computer Science, 2014.

[32] Jan Křetínský and Salomon Sickert. MoTraS: A tool for modal transition systems and their extensions. In Dang Van Hung and Mizuhito Ogawa, editors, *ATVA*, volume 8172 of *Lect. Notes Comput. Sci.*, pages 487–491. Springer-Verlag, 2013. Tool accessible at `https://www7.in.tum.de/~kretinsk/motras.html`.

[33] Kim G. Larsen. Modal specifications. In Joseph Sifakis, editor, *Automatic Verification Methods for Finite State Systems*, volume 407 of *Lect. Notes Comput. Sci.*, pages 232–246. Springer-Verlag, 1989.

[34] Kim G. Larsen. Ideal specification formalism = expressivity + compositionality + decidability + testability + ... In Jos C. M. Baeten and Jan Willem Klop, editors, *CONCUR*, volume 458 of *Lect. Notes Comput. Sci.*, pages 33–56. Springer-Verlag, 1990.

[35] Kim G. Larsen. Proof systems for satisfiability in Hennessy-Milner logic with recursion. *Theor. Comput. Sci.*, 72(2&3):265–288, 1990.

[36] Kim G. Larsen and Liu Xinxin. Compositionality through an operational semantics of contexts. In Mike Paterson, editor, *ICALP*, volume 443 of *Lect. Notes Comput. Sci.*, pages 526–539. Springer-Verlag, 1990.

[37] Kim G. Larsen and Liu Xinxin. Equation solving using modal transition systems. In *LICS*, pages 108–117. IEEE Computer Society, 1990.

[38] Covering of a partial order by upwards convex sets. Mathoverflow discussion. `http://mathoverflow.net/questions/118031/`.

[39] Ulrik Nyman. *Modal Transition Systems as the Basis for Interface Theories and Product Lines*. PhD thesis, Institut for Datalogi, Aalborg Universitet, 2008.

[40] Peter W. O'Hearn, John C. Reynolds, and Hongseok Yang. Local reasoning about programs that alter data structures. In Laurent Fribourg, editor, *CSL*, volume 2142 of *Lect. Notes Comput. Sci.*, pages 1–19. Springer-Verlag, 2001.

[41] Vaughan R. Pratt. A decidable mu-calculus: Preliminary report. In *FOCS*, pages 421–427. IEEE Computer Society, 1981.

[42] Arthur N. Prior. *Papers on Time and Tense*. Oxford: Clarendon Press, 1968.

[43] Jean-Pierre Queille and Joseph Sifakis. Specification and verification of concurrent systems in CESAR. In Mariangiola Dezani-Ciancaglini and Ugo Montanari, editors, *Symp. Program.*, volume 137 of *Lect. Notes Comput. Sci.*, pages 337–351. Springer-Verlag, 1982.

[44] Jean-Baptiste Raclet. Residual for component specifications. Publication interne 1843, IRISA, Rennes, 2007.

[45] Jean-Baptiste Raclet. Residual for component specifications. *Electr. Notes Theor. Comput. Sci.*, 215:93–110, 2008.

[46] Jean-Baptiste Raclet, Eric Badouel, Albert Benveniste, Benoît Caillaud, and Roberto Passerone. Why are modalities good for interface theories? In *ACSD*, pages 119–127. IEEE Computer Society, 2009.

[47] John C. Reynolds. Separation logic: A logic for shared mutable data structures. In *LICS*, pages 55–74. IEEE Computer Society, 2002.

[48] Mathieu Sassolas, Marsha Chechik, and Sebastián Uchitel. Exploring inconsistencies between modal transition systems. *Software and System Modeling*, 10(1):117–142, 2011.

[49] Dana Scott and Jaco W. de Bakker. A theory of programs. Unpublished manuscript, IBM, Vienna, 1969.

[50] Sebastián Uchitel and Marsha Chechik. Merging partial behavioural models. In Richard N. Taylor and Matthew B. Dwyer, editors, *SIGSOFT FSE*, pages 43–52. ACM, 2004.

[51] Morgan Ward and R.P. Dilworth. Residuated lattices. *Trans. AMS*, 45(3):335–354, 1939.