Topological analysis of refinement

Michael Huth [1]

Department of Computing
Imperial College London
London, United Kingdom

Abstract

A modal transition system has a class of implementations, its maximal refinements. This class determines satisfiability and validity judgments, and their compositional approximations, for formulas of Hennessy-Milner logic. Using topology, we prove structural properties of these judgments: refinement is reverse containment of classes of implementations, Hennessy-Milner logic characterizes refinement through validity judgments, implementation classes are topologically closed sets, Hennessy-Milner logic enjoys a compactness theorem on such classes, and a robust consistency measure between modal transition systems is definable. In particular, every formula of Hennessy-Milner logic is the finite disjunction of Hennessy-Milner logic formulas for which validity checks are reducible to model checks.

Key words: topology, refinement, validity, model checking, modal transition system

1 Introduction

1.1 Motivation for writing this paper

This paper is loosely based on an invited talk given at the Third Irish Conference on the Mathematical Foundations of Computer Science and Information Technology at Trinity College, Dublin, Ireland, 22-23 July 2004. All technical results discussed in this paper have been published in [HJS01,HJS04,Hut04a] or have already been submitted elsewhere [Hut04b]. However we feel that there is value in discussing the research programme that underlies and unifies those publications as it creates an opportunity to present fairly technical matters in an informal manner and with focus on conceptual points. Such an exposition also enables us to present open research problems in a readily accessible and, hopefully, well motivated way.

[1] Email: M.Huth@doc.imperial.ac.uk

This is a preliminary version. The final version will be published in Electronic Notes in Theoretical Computer Science
URL: www.elsevier.nl/locate/entcs

1.2 Motivation for reported research

Topology may be seen as a methodology for extending mathematical techniques and insights from finite sets and functions to the infinite, where the constraints of, and particularities for, such extensions are the subject of study. This biased view of topology is appropriate for our object of study here.

Refinements are understood to be binary relations on classes of mathematical models. An implementation relation between models and their allowed implementations is subsumed by this understanding in relating models M to all their refinements I that are "maximal" (i.e. all refinements of I are refinement-equivalent to I). The need to refine models can be found in many areas of computer science. We limit ourselves to two examples, both of which are covered by the technical refinement notion discussed in this paper.

Example 1.1 (i) **Program abstraction:** *From an operational semantics of a program P one may construct a mathematical model of state and behavior M [Plo81]. Since M may have infinitely many or "too many" states, we may want to abstract M into a "smaller" model A and analyze A instead. The model M should refine A when all behavior possible in M is also possible in A, which is then a* safe simulation *of M [CC77,BPR01].*

(ii) **Specification:** *Designs distinguish themselves from implementations in that certain aspects of state or behavior are not yet decided or known. A component in the Microsoft .NET framework, e.g., may have a* `main` *method, which would start the execution, but it may not have such a method if the component is an audio plug-in for a web browser [EJS03]. Any instance of a component will either have a* `main` *method or it won't. But the "design template" for components won't prescribe whether such a method is present. Implementations should be refinements of designs such that all optional or under-specified aspects have been determined.*

This paper demonstrates that topology is useful for proving structural properties about refinement as it enables us to prove those properties for refinement restricted to a certain class of finite-state models. The insights gained by this topological analysis of refinement then have potentially important applications and consequences of which we mention consistency measures between two models, a possible reduction of satisfiability checks on implementation classes to model checks, and a compactness theorem for Hennessy-Milner logic and *common* implementations of finitely many models.

The work presented here can also be seen as a unification of related strands of work:

- a metric semantics of processes by de Bakker & Zucker [dBZ82], which solves recursive equations over complete metric spaces,
- a framework for under-specification and refinement of systems, which is proposed by Larsen & Thomsen in [LT88],
- domain theory for modelling transition systems and their refinement, which

is worked out for partial bisimulation by Abramsky in [Abr91], and

- the representation of classical topological spaces as maximal-points spaces of a domain, which is pioneered by Lawson in [Law97].

This unification, and the topological means that it provides, can then be exploited to determine insights into the structure of refinement, notably

- a compactness theorem for temporal logic,
- a consistency measure for under-specification,
- the realization of refinement as inverse containment of implementations, and
- the collective model checking of multiple models.

We sketch the nature and proof of these insights but won't say much about the last item for which [HH04] contains further details.

1.3 Outline of paper

In Section 2 we present the models studied in this paper, along with their refinement notion, and mention that these models are quite expressive. In Section 3 we discuss the fully abstract and universal domain model for refinement, as developed in [HJS04], and motivate it through a comparison with the interval domain. In Section 4 we show that the set of maximal points of our domain model, equipped with the topology induced by the Scott-topology, is a Stone space and model of the set of bisimulation equivalence classes. This then leads to a compactness theorem for Hennessy-Milner logic on sets of implementations of models. Measures of consistency between two models are defined and studied in Section 5, where compactness of implementation sets allows us to prove the robustness of these notions. In Section 6 we sketch a proof that refinement between models is nothing but reserve containment of the respective sets of implementations: a model M refines a model N iff all implementations of M are also implementations of N. We therefore realize that validity checks are model checks for a certain set of formulas of Hennessy-Milner logic that generates the entire logic through disjunction. Finally, Section 7 concludes. The papers [HJS01,HJS04,Hut04a,Hut04b,HH04] should be consulted for accounts of related work.

2 Modal transition systems and refinement

In this section we define modal transition systems, their co-inductive refinement, point out the expressiveness of this formalism, and give an approximating semantics of temporal logic which characterizes refinement.

2.1 Models

Example 2.1 [[Hut04a]] Figure 1 shows a modal transition system of "pub behavior" [Hut04a]. From state 'Waits' it is guaranteed that event 'newpint'

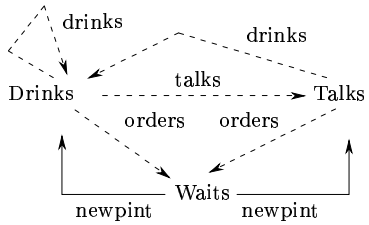

Fig. 1. A modal transition system specifying pub behavior [Hut04a]. Solid lines denote R^a-transitions, dashed lines denote $R^c \setminus R^a$-transitions.

may lead to any of the other two states, as indicated by the two solid transitions from 'Waits.' The dashed transition from 'Drinks' to 'Talks' labelled with event 'talks' indicates that an implementation could have this reactive ability but is not guaranteed to possess it. Its implementation is at the discretion of the implementor.

Throughout this paper, we fix a finite set of events Act. A modal transition system M is a tuple

$$(\Sigma; R^a, R^c \subseteq \Sigma \times Act \times \Sigma) \tag{1}$$

- where Σ is the state space, e.g. $\{\text{Drinks}, \text{Talks}, \text{Waits}\}$ in Figure 1,
- R^a consists of all solid lines, the must-transitions in [LT88],
- $R^c \setminus R^a$ comprises all dashed lines, the may-transitions in [LT88],
- $\Sigma \times Act \times \Sigma \setminus R^c$ denotes all "absent" lines, and
- there is a consistency condition on transition relations: $R^a \subseteq R^c$.

As a specification, a modal transition system M has two kinds of *contractual guarantees*: elements of R^a are guaranteed to be implemented whereas elements of $\Sigma \times Act \times \Sigma \setminus R^c$ are guaranteed not to be implemented. The latter is an indirect guarantee since R^c is the universe of *contractually possible behavior*. These guarantees have to be interpreted with respect to refinement. For example, $(\text{Drinks}, \text{newpint}, \text{Talks}) \notin R^c$ in Figure 1 means that event 'newpint' cannot lead from any state refining 'Drinks' to a state that refines 'Talks.'

2.2 Refinement

This understanding of guarantees and possibilities suggests a co-inductive definition of refinement [Lar89,Dam96,DGG97]. A relation $Q \subseteq \Sigma \times \Sigma$ is a *refinement* for a modal transition system as in (1) iff $(s,t) \in Q$ implies

(i) if $(s, \alpha, s') \in R^a$, there is $(t, \alpha, t') \in R^a$ with $(s', t') \in Q$

(ii) if $(t, \alpha, t') \in R^c$, there is $(s, \alpha, s') \in R^c$ with $(s', t') \in Q$.

So t *refines* s (or equivalently, s *abstracts* t) iff $(s,t) \in Q$ for some refinement relation Q. We say that s and t are *refinement-equivalent* if each one refines the other.

Example 2.2 [[Hut04a]] On the right of Figure 2 we see a refinement of the

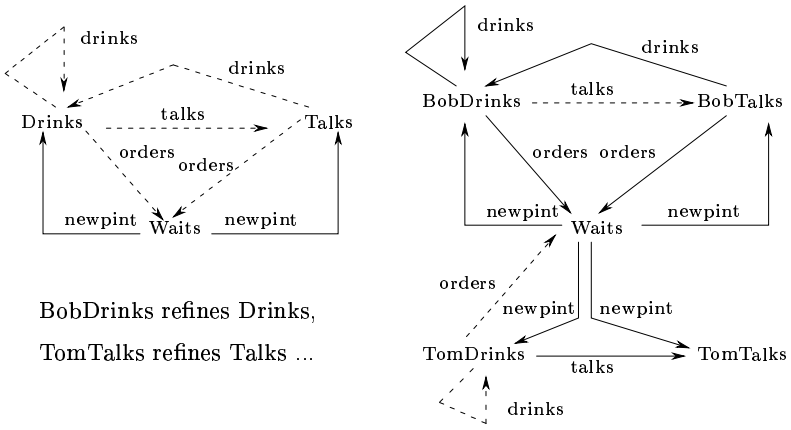

Fig. 2. To the left: the modal transition system of Figure 1. To the right: a refinement of this modal transition system, witnessed by relation Q.

modal transition system from Figure 1. The relation

$$Q = \{(\text{Drinks}, \text{BobDrinks}), (\text{Drinks}, \text{TomDrinks}), (\text{Waits}, \text{Waits}), (\text{Talks}, \text{BobTalks}), (\text{Talks}, \text{TomTalks})\}$$

is a refinement in the modal transition system that is the union of the systems to the left and right of Figure 2.

Modal transition systems are event-based so modelling and analysis of models seems to be limited to the notion of events. But this is not so. Godefroid & Jagadeesan [GJ03] have shown that most 3-valued formalisms used in practice inter-translate in PTIME and LOGSPACE such that the translations of models and temporal logic formulas preserve and reflect refinement and the results of model checks. Two prominent such formalisms are

- the *partial Kripke structures* of Bruns & Godefroid [BG99], which are tuples $(\Sigma; R \subseteq \Sigma \times \Sigma; L^a, L^c \colon AP \to \mathcal{P}(\Sigma))$ where we have 2-valued transitions $(s, s') \in R$, and 3-valued state propositions $L^a(q) \subseteq L^c(q)$ for each $q \in AP$, and
- *Kripke modal transition systems* [HJS01], which are tuples $(\Sigma; R^a, R^c \subseteq \Sigma \times Act \times \Sigma; L^a, L^c \colon AP \to \mathcal{P}(\Sigma))$ where we have 3-valued transitions $R^a \subseteq R^c$ and 3-valued state propositions $L^a(q) \subseteq L^c(q)$ for each $q \in AP$.

In these models, $s \in L^a(q)$ means that "q is asserted at s" whereas $s \in L^c(q)$ means that "q is consistent at s." The result in [GJ03] therefore ensures that our domain model and the topological insights presented subsequently capture all such refinement formalisms, be they event-based, state-based or both.

2.3 Temporal logic

Creating a model of a specification can be an important part of design. Given such a model, we may wish to analyze it in order to debug or certify de-

signs. We use a simple temporal logic, Hennessy-Milner logic [HM85], to that end. Although this logic does not have fixed points, it is expressive enough for bounded testing, validation, and simulation activities. The syntax of Hennessy-Milner logic is given by the grammar

$$\phi ::= tt \mid \neg\phi \mid \langle\alpha\rangle\phi \mid \phi \wedge \phi \tag{2}$$

where α ranges over a finite set of events Act. There are two kinds of judgments:

- $s \models^a \phi$, which intends to say that "ϕ is **a**sserted at s," and
- $s \models^c \phi$, which intends to say that "ϕ may be **c**onsistent at s."

The first judgment under-approximates the *validity judgment*

$$\mathbb{V}(s,\phi) = \{\text{all implementations of } s \text{ satisfy } \phi\}\,. \tag{3}$$

and the second one over-approximates the dual *satisfiability judgment*

$$\mathbb{S}(s,\phi) = \{\text{some implementation of } s \text{ satisfies } \phi\} \tag{4}$$

We return to this loss of precision and the possibility of its mitigation at the end of this paper. The semantics of the judgments $\models^a$ and $\models^c$ are compositional and given by

- $s \models^m tt$
- $s \models^m \neg\phi$ iff (not $s \models^{\neg m} \phi$, where $\neg a = c$ and $\neg c = a$)
- $s \models^m \langle\alpha\rangle\phi$ iff (for some $(s,\alpha,s') \in R^m$, $s' \models^m \phi$)
- $s \models^m \phi_1 \wedge \phi_2$ iff ($s \models^m \phi_1$ and $s \models^m \phi_2$)

where $m \in \{a, c\}$. Disjunction $\phi \vee \psi$ is an abbreviation of $\neg(\neg\phi \wedge \neg\psi)$ and the box modality $[\alpha]$ denotes $\neg\langle\alpha\rangle\neg$ for each $\alpha \in Act$. From the definitions above we can read off the semantics of disjunction and the box modality:

- $s \models^m \phi_1 \vee \phi_2$ iff ($s \models^m \phi_1$ or $s \models^m \phi_2$), and
- $s \models^m [\alpha]\phi$ iff (for all $(s,\alpha,s') \in R^{\neg m}$, $s' \models^m \phi$).

Please note that the universal quantifier for the box modality ranges over transitions in a mode *dual* to that of the judgment. For example if we check $s \models^a [\alpha]tt$, we need to look at all R^c-transitions out of s labelled with α.

Example 2.3 [[Hut04a]] Reconsider Figure 1.

(i) We have $\text{Talks} \models^c \langle\text{drinks}\rangle tt$ as $(\text{Talks}, \text{drinks}, \text{Drinks}) \in R^c$. Therefore $\text{Talks} \not\models^a \neg\langle\text{drinks}\rangle tt$. We also have $\text{Talks} \not\models^a \langle\text{drinks}\rangle tt$ as there is no x with $(\text{Talks}, \text{drinks}, x) \in R^a$. Therefore $\text{Talks} \not\models^a \langle\text{drinks}\rangle tt \vee \neg\langle\text{drinks}\rangle tt$ despite the fact that $\langle\text{drinks}\rangle tt \vee \neg\langle\text{drinks}\rangle tt$ is a tautology over labelled transition systems.

(ii) We also have $\text{Waits} \not\models^a [\text{newpint}][\text{talks}](\langle\text{drinks}\rangle tt \vee \neg\langle\text{drinks}\rangle tt)$ as:
- $(\text{Waits}, \text{newpint}, \text{Drinks})(\text{Drinks}, \text{talks}, \text{Talks})$ is an R^c-path recognizing the word 'newpint talks,' and
- $\text{Talks} \not\models^a \langle\text{drinks}\rangle tt \vee \neg\langle\text{drinks}\rangle tt$ by item (i).

The intuition conveyed by this example is that a modal transition system M "is" a labelled transition system iff M passes all tests

$$[\delta_1][\delta_2]\ldots[\delta_n](\langle\alpha\rangle\phi_k \vee \neg\langle\alpha\rangle\phi_k) \tag{5}$$

for $\alpha, \delta_i \in Act$ and suitable ϕ_k of Hennessy-Milner logic in the $\models^a$ semantics. Suitability means that the set of ϕ_k has to be large enough to characterize refinement. For example, we could choose all formulas of Hennessy-Milner logic as such ϕ_k since Larsen has shown [Lar89] that, with our notation, the following statements are equivalent for states s and t of modal transition systems:

(i) state t refines state s

(ii) for all ϕ of Hennessy-Milner logic, $s \models^a \phi$ implies $t \models^a \phi$

(iii) for all ϕ of Hennessy-Milner logic, $t \models^c \phi$ implies $s \models^c \phi$.

This result generalizes a corresponding result for bisimulation since, for labelled transition systems, refinement is bisimulation and $\models^a$ equals $\models^c$ and is the familiar semantics of Hennessy-Milner logic over labelled transition systems where a labelled transition system (Σ, R) is interpreted as a modal transition system (Σ, R, R). Another important consequence of this logical characterization is that judgments $s \models^a \phi$ are sound under refinements of s, item (ii) above, whereas judgments $t \models^c \phi$ are sound under *abstractions* of t, item (iii) above, where an abstraction of t is a state u such that t refines u.

3 Domain model for refinement

3.1 The interval domain as a metaphor

In this section we use Scott's interval domain [Sco72] $\mathbb{I}$ as a metaphor to set the stage for a domain model $\mathbb{D}$ of modal transition systems developed in [HJS04]. The reader may wish to consult the brief Appendix A for basic terminology and notation from domain theory [AJ94]. The domain $\mathbb{D}$ has a natural interpretation as a modal transition system $\mathcal{D}$ such that $\mathcal{D}$ is *universal*: all modal transition systems have refinement-equivalent embeddings in $\mathcal{D}$. The domain $\mathbb{D}$ is also *fully abstract*: the order on the domain $\mathbb{D}$ is the greatest refinement relation in $\mathcal{D}$, which is the union of all refinements in $\mathcal{D}$. These two properties, along with the structure of two topologies on $\mathbb{D}$, are key ingredients for securing the results presented in this paper.

Example 3.1 Figure 3 shows the interval domain and its ordering: $[r, s] \leq [r', s']$ iff $(r \leq r'$ and $s' \leq s)$. In that case we say that $[r', s']$ refines $[r, s]$.

The interval domain nicely illustrates some of the properties we expect our domain model for refinement $\mathbb{D}$ to have.

(i) **Refinement is complete for implementations:** If real numbers $x \in [0, 1]$ represented as intervals $[x, x]$ are seen as the possible implementations of intervals, then $[r, s]$ has $\{[x, x] \mid x \in [r, s]\}$ as set of implemen-

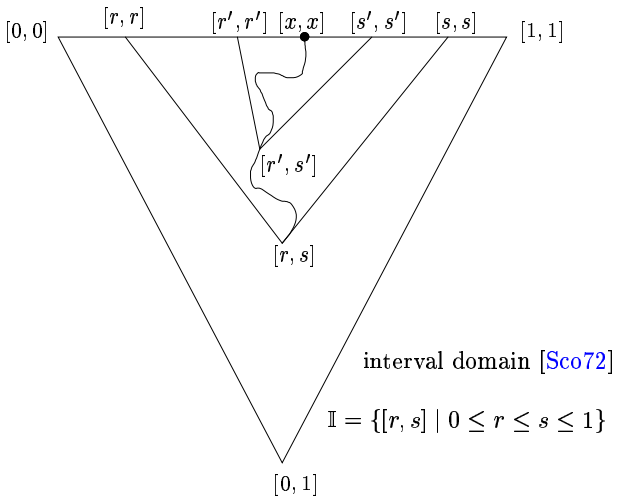

Fig. 3. A schematic description of the interval domain and its order: $[r, s] \leq [r', s']$ iff ($r \leq r'$ and $s' \leq s$).

tations. Since we may identify that set with $[r, s]$, it is therefore evident that $[r, s]$ is refined by $[r', s']$ iff all implementations of $[r', s']$ are also implementations of $[r, s]$.

(ii) **Universality:** The interval domain $\mathbb{I}$ is universal for worst/best-case abstractions of subsets of $[0, 1]$. If we abstract $X \subseteq [0, 1]$ with the interval $[\bigwedge X, \bigvee X]$, the latter is in $\mathbb{I}$ and any element of $\mathbb{I}$ is the abstraction of at least one such X.

(iii) **Full abstraction:** The order on $\mathbb{I}$ exactly captures the refinement relation if the latter means reverse containment of implementations, which is does by virtue of item (i) above.

(iv) **Classical space as maximal-points space:** The set $[0, 1]$ equipped with the Euclidean topology is isomorphic as a topological space to the set of maximal elements of $\mathbb{I}$ in the topology induced by the Scott- or Lawson-topology of $\mathbb{I}$.

(v) **Denseness of computable structures:** Intervals with rational endpoints approximate intervals to any degree of precision within $\mathbb{I}$.

(vi) **Consistency measure:** The map $c\colon \mathbb{I} \times \mathbb{I} \to \mathbb{I}_\perp$ given by $c([r, s], [r', s']) = [\max(r, r'), \min(s, s')]$, where $[x, y]$ is understood to be $\perp$ if $x \not\leq y$, tells us whether its inputs are consistent with each other by checking whether its output is different from $\perp$.

3.2 The domain model of [HJS04]

Our objectives are to devise a domain model $\mathbb{D}$ for modal transition systems and their refinement with enjoys similar facts for labelled transition systems as elements of $\max(\mathbb{D})$. In particular we wish to secure that refinement is complete and that there is a meaningful, monotone consistency measure $c\colon \mathbb{D} \times$

$\mathbb{D} \to \mathbb{I}$. The domain model $\mathbb{D}$ [HJS04] is obtained as the initial, ω-algebraic, and bifinite solution $\mathbb{D}$ of

$$D = \prod_{\alpha \in Act} \mathcal{M}[D] \tag{6}$$

where $\mathcal{M}[D]$ is the mixed powerdomain of Heckmann & Gunter [Hec90,Gun92]. Elements of $\mathcal{M}[D]$ are all pairs (L, U) where $L = \downarrow L$ and $U = \uparrow U$ are Lawson-closed, and (L, U) satisfies the mix condition

$$L = \downarrow\!(L \cap U)\,. \tag{7}$$

The order on $\mathcal{M}[D]$ is given by

$$(L, U) \leq (L', U') \text{ iff } (L \subseteq L' \;\&\; U' \subseteq U) \tag{8}$$

and so the L sets grow and the U sets shrink in increasing sequences in $\mathcal{M}[D]$.

Example 3.2 [[Hut04a]] Here are some elements of $\mathbb{D}$:

- $\bot_{\mathbb{D}} = (\{\}, \mathbb{D})_{\alpha \in Act} \in \mathbb{D}$ models a *"universal stub"* that refines every modal transition system over Act, and
- $(\{\}, \{\})_{\alpha \in Act} \in \max(\mathbb{D})$ models *deadlock* as it cannot possibly engage in any events; this is essentially a labelled transition system.

Condition (7) turns out to be an ordered version of $R^a \subseteq R^c$ since we can make $\mathbb{D}$ into a modal transition system [HJS04]. The idea behind this is that sets L encode the set of R^a-successors and U the set of R^c-successors, sorted by events as in the recursion (6). The recursion pattern

$$d = ((d^a_\alpha, d^c_\alpha))_{\alpha \in Act} \tag{9}$$

derived from (6) defines a tuple $\mathcal{D} = (\mathbb{D}; \mathbb{R}^a, \mathbb{R}^c)$ where

$$\begin{aligned} \mathbb{R}^a &= \{(d, \alpha, d') \in \mathbb{D} \times Act \times \mathbb{D} \mid d' \in d^a_\alpha\} \\ \mathbb{R}^c &= \{(d, \alpha, d') \in \mathbb{D} \times Act \times \mathbb{D} \mid d' \in d^c_\alpha\} \end{aligned} \tag{10}$$

and so d^a_α (d^c_α) is the set of $\mathbb{R}^a_\alpha$-successors ($\mathbb{R}^c_\alpha$-successors) of d (respectively) for each $\alpha \in Act$. A minor detail is noteworthy: we have $\mathbb{R}^a \not\subseteq \mathbb{R}^c$ and so $\mathcal{D}$ is *not* a modal transition system. But $\mathcal{D}$ is refinement-equivalent to the *modal* transition system $(\mathbb{D}; \mathbb{R}^a \cap \mathbb{R}^c, \mathbb{R}^c)$ by virtue of (7) if we apply the same definition of refinement as before to systems that may not satisfy $R^a \subseteq R^c$ [HJS04]. For $\mathbb{D}$ we are therefore entitled to write $\mathcal{D}$ whenever we mean the modal transition system $(\mathbb{D}; \mathbb{R}^a \cap \mathbb{R}^c, \mathbb{R}^c)$.

Remark 3.3 *We do not know whether systems $(\Sigma; R^a, R^c)$ that don't satisfy $R^a \subseteq R^c$ are refinement-equivalent to some modal transition systems whenever $(\Sigma; R^a, R^c)$ has some labelled transition system as a refinement. That is to say, we do not know whether the mix condition (7) is also necessary (not just sufficient) for non-emptiness of implementation classes.*

The universality of $\mathcal{D}$ [HJS04] states that for any modal transition system M with initial state i there is $\langle\!|\, M, i \,|\!\rangle \in \mathbb{D}$ such that (M, i) and $(\mathcal{D}, \langle\!|\, M, i \,|\!\rangle)$ are

$$\{\!|\ \mathbf{0}\ |\!\} = ((\{\},\{\}))_{\alpha\in Act}$$
$$\{\!|\ \bot\ |\!\} = \bot_{\mathbb{D}}$$
$$(\{\!|\ \alpha_{tt}.p\ |\!\}^a_\alpha, \{\!|\ \alpha_{tt}.p\ |\!\}^c_\alpha) = (\downarrow\{\!|\ p\ |\!\}, \uparrow\{\!|\ p\ |\!\})$$
$$(\{\!|\ \alpha_{\bot}.p\ |\!\}^a_\alpha, \{\!|\ \alpha_{\bot}.p\ |\!\}^c_\alpha) = (\{\}, \uparrow\{\!|\ p\ |\!\})$$
$$(\{\!|\ \alpha_{v}.p\ |\!\}^a_\beta, \{\!|\ \alpha_{v}.p\ |\!\}^c_\beta) = (\{\},\{\}),\ \alpha\neq\beta,\ v\in\{tt,\bot\}$$
$$\{\!|\ p+q\ |\!\}^m_\gamma = \{\!|\ p\ |\!\}^m_\gamma \cup \{\!|\ q\ |\!\}^m_\gamma,\ \gamma\in Act,\ m\in\{a,c\}$$

Fig. 4. Denotational semantics of process terms in $\mathbb{D}$. It interprets **0** as deadlock, $\bot$ as universal stub, $+$ as mix union of [Hec90], and prefixes as expected (plus saturations with $\downarrow$ and $\uparrow$).

refinement-equivalent,[2] where we write (M,i) for a modal transition system M with initial state i and where refinement means refinement of initial states by initial states. This result can be proved as follows, for a full proof please see [HJS04]:

(i) for each $n\geq 0$ unwind and truncate (M,i) as a tree of depth $\leq n$,

(ii) express truncations as denotations of terms in a 3-valued process algebra with grammar

(11) $$p ::= \mathbf{0} \mid \bot \mid \alpha_{tt}.p \mid \alpha_{\bot}.p \mid p+p \quad (\alpha\in Act)$$

where none of the p in clause $p+p$ are allowed to be $\bot$,

(iii) realize (M,i) as a "refinement limit" of truncations,

(iv) embed truncations p into $\mathbb{D}$ through the denotational semantics of process algebra terms in Figure 4, and finally

(v) use a continuity/compactness argument in $\mathbb{D}$ for refinement equivalence.

The denotations of process terms in $\mathbb{D}$ are given in Figure 4 for sake of illustration.

Example 3.4 [[Hut04a]] Figure 5 shows an example truncation of depth one for the state 'TomDrinks' from Figure 2. We observe two kinds of leaves: universal stubs, here for 'TomDrinks' and 'Waits,' and deadlock, here for 'TomTalks.'

Next, one can prove full abstraction of $\mathbb{D}$ [HJS04] saying that the order on $\mathbb{D}$ is the greatest refinement relation on $\mathcal{D}$, which is the union of all refinement relations: for all $d,e\in\mathbb{D}$, we have $d\leq e$ iff $(\mathcal{D},e)$ refines $(\mathcal{D},d)$. To prove this, we

(i) show that the order $\leq$ on $\mathbb{D}$ is a refinement within $\mathcal{D}$, which is hardwired into the definitions of $\mathbb{D}$ and $\mathcal{D}$, and

[2] This fact is proved in [HJS04] for *image-finite* models, where for all $s\in\Sigma$, $\alpha\in Act$, and $m\in\{a,c\}$ the set $\{s'\in\Sigma \mid (s,\alpha,s')\in R^m\}$ is finite.

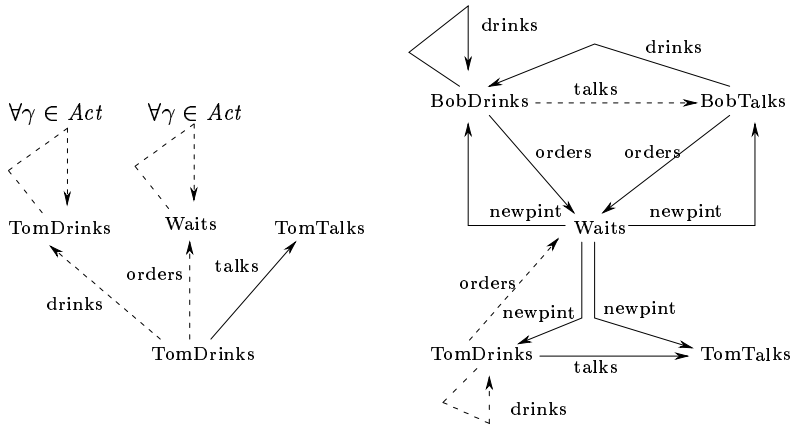

Fig. 5. To the left: Truncation of depth one for 'TomDrinks' from Figure 2 (to the right). We observe two kinds of leaves: universal stubs and deadlock.

(ii) use the logical characterization of refinement to show that $d \not\leq e$ implies that $(\mathcal{D}, e)$ does not refine $(\mathcal{D}, d)$:
   (a) $\mathbf{K}(\mathbb{D})$, the set of compact elements of $\mathbb{D}$, order-generates $\mathbb{D}$ as the latter is algebraic so $d \not\leq e$ implies $k \leq d$ and $k \not\leq e$ for some $k \in \mathbf{K}(\mathbb{D})$
   (b) for each $k \in \mathbf{K}(\mathbb{D})$ there is a Hennessy-Milner formula ϕ_k so that for all $f \in \mathbb{D}$: $k \leq f$ iff $f \models^a \phi_k$
   (c) thus $d \models^a \phi_k$ and $e \not\models^a \phi_k$ imply that e does not refine d in $\mathcal{D}$ by item (ii) on page 7.

Again, we refer for further details of this proof to [HJS04].

4 Compactness theorem for refinement

4.1 Topological space of maximal points

In this section we define the terms Scott-topology, Lawson-topology, and Stone space, define the induced topology $\tau_{\mathbb{X}}$ on $\mathbb{X} = \max(\mathbb{D})$, and outline a proof of the fact that $(X, \tau_{\mathbb{X}})$ is a Stone space and the quotient space of labelled transition systems modulo bisimulation. The latter then gives us a compactness theorem for implementation classes. We write

$$\mathbb{X} = \max(\mathbb{D}) = \{d \in \mathbb{D} \mid \forall e \in \mathbb{D}: d \leq e \Rightarrow d = e\} \tag{12}$$

for the set of maximal elements of $\mathbb{D}$ and require three topologies here.

(i) The *Scott-topology* of $\mathbb{D}$ is

$$\sigma_{\mathbb{D}} = \{\uparrow k \mid k \in \mathbf{K}(\mathbb{D})\} \tag{13}$$

and is T_0 such that $\mathbf{K}(\mathbb{D})$ is the set of embeddings of all truncated, image-finite trees [HJS04] (or, equivalently, the set of meanings $\{\!|\, p \,|\!\}$ for all process terms p of (11)).

(ii) The *Lawson-topology* is

$$\lambda_{\mathbb{D}} = \{\uparrow k \setminus \uparrow l \mid k, l \in \mathbf{K}(\mathbb{D})\} \tag{14}$$

and is compact Hausdorff [AJ94].

(iii) Finally the *Lawson-condition*, a consistency condition between the Scott- and Lawson-topology on $\max(\mathbb{D})$, is crucial: the topology

(15) $$\tau_{\mathbb{X}} = \{U \cap \max(\mathbb{D}) \mid U \in \sigma_{\mathbb{D}}\}$$

on $\mathbb{X}$ equals $\{V \cap \max(\mathbb{D}) \mid V \in \lambda_{\mathbb{D}}\}$ on $\mathbb{X}$ as $\mathbb{D}$ is bifinite [Law97].

Let us recall that $(\mathbb{X}, \tau_{\mathbb{X}})$ is a *Stone space* [Joh82] iff $\tau_{\mathbb{X}}$ is compact, Hausdorff, and zerodimensional where

- *compact* means: for all $\mathcal{U} \subseteq \tau_{\mathbb{X}}$ with $\mathbb{X} \subseteq \bigcup \mathcal{U}$ there is a finite $\mathcal{F} \subseteq \mathcal{U}$ with $\mathbb{X} \subseteq \bigcup \mathcal{F}$
- *Hausdorff* means: for all $x \neq x'$ in $\mathbb{X}$ there are $O, O' \in \tau_{\mathbb{X}}$ with $x \in O$, $x' \in O'$, and $O \cap O' = \{\}$, and
- *zero-dimensional* means: every $U \in \tau_{\mathbb{X}}$ is the union of sets that are $\tau_{\mathbb{X}}$-open (i.e. elements of $\tau_{\mathbb{X}}$) *and* $\tau_{\mathbb{X}}$-closed (i.e. elements of the complement of $\tau_{\mathbb{X}}$).

4.2 Compactness of maximal-points space

Applying the Lawson condition one can see that $\uparrow\!k \cap \max(\mathbb{D}) = \{m \in \max(\mathbb{D}) \mid k \leq m\}$ is $\tau_{\mathbb{X}}$-open and $\tau_{\mathbb{X}}$-closed for each $k \in \mathbf{K}(\mathbb{D})$. Using this, it is easily proved that $\tau_{\mathbb{X}}$ is zero-dimensional and Hausdorff. As $\lambda_{\mathbb{D}}$ is compact, it suffices to show that $\max(\mathbb{D})$ is $\lambda_{\mathbb{D}}$-closed (and therefore $\tau_{\mathbb{X}}$-compact). This is where we make use of a complete set of tests for maximality. We write Act^* for the set of finite words over the finite set of events Act.

- For $\Delta = \delta_1\delta_2 \ldots \delta_n \in Act^*$, $\alpha \in Act$, and $k \in \mathbf{K}(\mathbb{D})$ we define the test formula $\psi_k^{\Delta,\alpha} = [\delta_1][\delta_2] \ldots [\delta_n](\langle\alpha\rangle\phi_k \vee \neg\langle\alpha\rangle\phi_k)$ of Hennessy-Milner logic where

  (16) $$\text{for all } d \in \mathbb{D} \text{ we have } (\mathcal{D}, d) \models^a \phi_k \text{ iff } k \leq d$$

  as stated earlier.

- For $m \in \{a, c\}$ and ϕ a formula of Hennessy-Milner logic, we write

  (17) $$[\![\, \phi \,]\!]^m = \{d \in \mathbb{D} \mid (\mathcal{D}, d) \models^m \phi\}\,.$$

- We form the set of points that pass all tests in the $\models^a$ semantics:

  (18) $$C = \bigcap\{[\![\, \psi_k^{\Delta,\alpha} \,]\!]^a \mid \Delta \in Act^*, \alpha \in Act, k \in \mathbf{K}(\mathbb{D})\}\,.$$

Our plan is then to show that

(i) for each formula ϕ of Hennessy-Milner logic, $[\![\, \phi \,]\!]^a$ is $\lambda_{\mathbb{D}}$-closed, and

(ii) $C = \max(\mathbb{D})$.

If this plan succeeds, $\max(\mathbb{D})$ is $\lambda_{\mathbb{D}}$-closed as the intersection of $\lambda_{\mathbb{D}}$-closed sets. For a full proof of both items above we refer to [Hut04a] and only sketch the structure of the arguments: we show that

(i) $[\![\, \phi \,]\!]^a$ is $\lambda_{\mathbb{D}}$-closed for all formulas ϕ of Hennessy-Milner logic by mutual

structural induction on ϕ in the stronger induction hypothesis

(19) "$\llbracket\ \phi\ \rrbracket^c$ and $\llbracket\ \phi\ \rrbracket^a$ are $\lambda_{\mathbb{D}}$-closed *and* $\lambda_{\mathbb{D}}$-open,"

(ii) $\max(\mathbb{D}) \subseteq C$; as C is $\lambda_{\mathbb{D}}$-closed, it suffices to show that the set of embeddings of labelled transition systems are in C and dense in $\max(\mathbb{D})$, and

(iii) $C \subseteq \max(\mathbb{D})$, by exploiting the fine structure of $(\mathbb{D}, \leq)$ and that $d \in C$ passes all tests $d \models^a \psi_k^{\Delta,\alpha}$.

In conclusion, $(\mathbb{X}, \tau_{\mathbb{X}})$ is a Stone space. We can furthermore demonstrate that this Stone space is the quotient space of labelled transition systems modulo bisimulation [Hut04a]:

(i) the embedding $(M, i) \mapsto \langle\!\langle\ M, i\ \rangle\!\rangle$ maps labelled transition systems with designated initial state into $\max(\mathbb{D})$,

(ii) any $(\mathcal{D}, d)$ with $d \in \max(\mathbb{D})$ is refinement-equivalent to a labelled transition system as then $e_\alpha^a \cap e_\alpha^c = e_\alpha^c \subseteq \max(\mathbb{D})$ for all $\alpha \in Act$ and e that are reachable from d via $\mathbb{R}^c$, and

(iii) we have a bijection of sets

(20) $$\mathbb{X} = \prod_{\alpha \in Act} \mathcal{C}[\mathbb{X}, \tau_{\mathbb{X}}]$$

where $\mathcal{C}[\mathbb{X}, \tau_{\mathbb{X}}]$ denotes the set of $\tau_{\mathbb{X}}$-compact subsets of $\mathbb{X}$, so x_α is the $\tau_{\mathbb{X}}$-compact set of α-successors for $x = (x_\alpha)_{\alpha \in Act} \in \mathbb{X}$.

Therefore our domain equation for modal transition systems renders a Stone space equation on the maximal-points space for $\mathbb{D}$.

4.3 A compactness theorem for temporal logic and refinement

We can now state a first consequence of these results, a compactness theorem for refinement.

Theorem 4.1 ([Hut04a]) *Let (M_k, i_k) be finitely many modal transition systems with respective initial states i_k and Γ a set of formulas of Hennessy-Milner logic such that for all finite subsets Π of Γ, the Hennessy-Milner logic formula $\bigwedge \Pi$ is satisfiable over labelled transition systems that refine all i_k. Then there is an image-finite labelled transition system (L, l) such that l refines all i_k and l satisfies all formulas of Γ.*

The proof of this theorem, which was formulated for a single model in [Hut04a] only, depends on the compactness of $\tau_{\mathbb{X}}$ and on the fact that each $\uparrow\!\langle\!\langle\ M_k, i_k\ \rangle\!\rangle \cap \max(\mathbb{D})$ is $\lambda_{\mathbb{D}}$-closed and so $\tau_{\mathbb{X}}$-closed.

Example 4.2 [[Hut04a]] If we have only one modal transition system (M_1, i_1) with $i_1 = \bot_{\mathbb{D}}$, then the theorem above is the familiar compactness theorem for Hennessy-Milner logic and labelled transition systems as all labelled transition systems refine $\bot_{\mathbb{D}}$.

5 Consistency measure for refinement

5.1 Two consistency measures

In this section we present metrics for modal and labelled transition systems, define two consistency measures for modal transition systems, and show that one of these is a robust optimistic measure of consistency. The two metrics are familiar from the literature. For any enumeration $k_0, k_1, \ldots$ of $\mathbf{K}(\mathbb{D})$ define

$$(21)\quad \begin{aligned} d_{\mathbb{D}}(d,e) &= inf\{2^{-n} \mid \forall i \leq n\colon k_i \leq d \text{ iff } k_i \leq e\} \\ d_{\mathbb{X}}(x,y) &= inf\{2^{-n} \mid \forall i \leq n\colon k_i \leq x \text{ iff } k_i \leq y\}\,. \end{aligned}$$

Noteworthy points about these metrics are that practical needs require an enumeration in increasing modal depth of ϕ_{k_n} for $n \geq 0$ so that in both metrics closer models require more effort (i.e. modal depth) for distinguishing them by tests. From standard domain and metric theory we learn that $d_{\mathbb{D}}$ induces $\lambda_{\mathbb{D}}$, and that $d_{\mathbb{X}}$ induces $\tau_{\mathbb{X}}$.

Let us say that, in a modal transition system, states

(22) s and t are *consistent* iff s and t have a common refinement.

Two consistency measures can now be defined [Hut04a]:

$$(23)\quad \begin{aligned} c_1(d,e) &= inf\{d_{\mathbb{X}}(x,y) \mid x \in \uparrow\! d \cap \max(\mathbb{D}),\ \ y \in \uparrow\! e \cap \max(\mathbb{D})\} \\ c_2(d,e) &= sup\,\{d_{\mathbb{X}}(x,y) \mid x \in \uparrow\! d \cap \max(\mathbb{D}),\ \ y \in \uparrow\! e \cap \max(\mathbb{D})\}\,. \end{aligned}$$

The intuition behind the measure c_1 is that we try to choose implementations of d and e, respectively, so as to minimize the distance between these implementations. Dually, in c_2 we seek to maximize that distance. The interval $[c_1(d,e), c_2(d,e)]$ is therefore an abstraction of the distance between d and e and the map

$$(24)\qquad (d,e) \mapsto [c_1(d,e), c_2(d,e)]\colon \mathbb{D} \times \mathbb{D} \to \mathbb{I}$$

is monotone. The degree of inconsistency between d and e can't be bigger than $c_2(d,e)$ nor smaller than $c_1(d,e)$.

5.2 Criteria for common refinements

From the compactness of $\tau_{\mathbb{X}}$ we infer that the measure c_1 is robust. Namely, we can show that

(25) $c_1(d,e) = 0$ iff d and e have a common refinement.

The latter is a decision problem in PTIME [HH04] but it may be harder to prove non-zero lower bounds for c_1. This is related to the fact that deciding bisimulation $\sim$ is in PTIME whereas Kanellakis & Smolka have shown that deciding bounded bisimulation $\sim_n$ is PSPACE-complete [KS90].

Example 5.1 Figure 6 schematically depicts the meaning of c_1 and c_2. Compactness ensures that if d and e have sequences of implementations $(x_i)_{i\geq 0}$ and $(y_i)_{i\geq 0}$, respectively, such that the limit of the sequence $(d_{\mathbb{X}}(x_i, y_i))_{i\geq 0}$ is 0, then d and e have a common refinement.

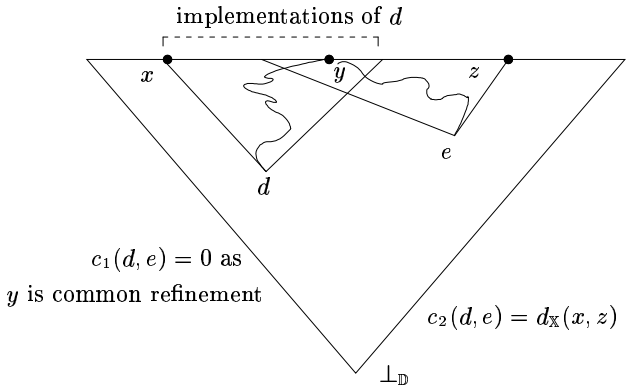

Fig. 6. Two modal transition systems d and e that have a common refinement.

6 Refinement is complete for implementations

6.1 Problem definition

In this section we prove soundness of refinement for implementations, sketch a proof for completeness of refinement for implementations, derive a new logical characterization for refinement, and discuss the connection between the completeness of refinement and the loss of precision for tests. For a detailed account of these insights and their full proofs we refer to [Hut04b].

The class of implementations

$$\mathcal{I}[M,s] = \{(L,l) \text{ labelled transition system} \mid (L,l) \text{ refines } (M,s)\} \tag{26}$$

of (M,s) is the class of all labelled transition systems (L,l) that refine (M,s). Since refinement is transitive we have $\mathcal{I}[N,t] \subseteq \mathcal{I}[M,s]$ whenever (N,t) refines (M,s). This implication captures *soundness* since it stipulates that step-wise refinement cannot introduce new implementations. The reverse implication ought to be true as well: reverse containment of implementations *ought to be* a refinement:

$$\text{Does } \mathcal{I}[N,t] \subseteq \mathcal{I}[M,s] \text{ imply that } (N,t) \text{ refines } (M,s)? \tag{27}$$

Example 6.1 Figure 7 illustrates the soundness and the putative incompleteness of refinement.

6.2 Proof sketch

One can prove the statement "for all modal transition systems (M,s) and (N,t), the inclusion $\mathcal{I}[N,t] \subseteq \mathcal{I}[M,s]$ implies that (N,t) refines (M,s)" by

(i) proving this for the case when s and t are denotations of process algebra terms generated by (11). This argument uses $\mathcal{I}[N,t] \subseteq \mathcal{I}[M,s]$ to dynamically synthesize winning strategies in refinement games for s and t, adapted from Stirling's work [Sti96] for bisimulation and labelled transition systems, and induction on the number of prefixed $\gamma_\perp$ with $\gamma \in Act$;

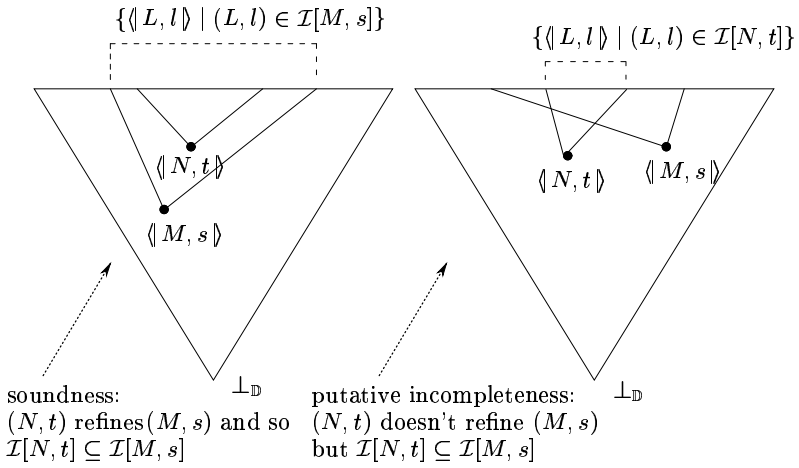

Fig. 7. To the left: an illustration of the soundness of refinement. To the right: an illustration of the putative incompleteness of refinement.

(ii) showing that "$\uparrow\!(\!|\, N, t \,|\!) \cap \max(\mathbb{D}) \subseteq \uparrow\!(\!|\, M, s \,|\!) \cap \max(\mathbb{D})$ implies $\mathcal{I}[N, t] \subseteq \mathcal{I}[M, s]$" for all (M, s) and (N, t); and

(iii) showing that "$\uparrow e \cap \max(\mathbb{D}) \subseteq \uparrow d \cap \max(\mathbb{D})$ implies $d \leq e$" for all $d, e \in \mathbb{D}$. This uses item (i), a compactness argument, and the fact that $\{d \in \mathbb{D} \mid \uparrow d \cap \max(\mathbb{D}) \subseteq \uparrow k\} \in \sigma_{\mathbb{D}}$ for $k \in \mathbf{K}(\mathbb{D})$ by the Hofman-Mislove Theorem [HM81].

6.3 Consequences of this proof

One consequence of this completeness is that refinement is also logically characterized by Hennessy-Milner logic for a semantics based on satisfiability or validity checks. From the soundness of $\models^a$ for refinement and of $\models^c$ for abstraction we get the implications

$$\begin{array}{lll} (28)\ s \models^a \phi & \Rightarrow & \mathbb{V}(s, \phi) \\ \quad s \models^c \phi & \Leftarrow & \mathbb{S}(s, \phi) \end{array}$$

for all states s of all modal transition systems and all formulas of Hennessy-Milner logic. But the converses of these implications are false in general. For example, all $\psi_k^{\Delta, \alpha}$ are tautologies over labelled transition systems and we saw that modal transition systems satisfy all such formulas with respect to $\models^a$ iff these systems are essentially labelled transition systems. We can now prove a novel logical characterization.

Theorem 6.2 ([Hut04b]) *Let t and s be states of modal transition systems. Then the following are all equivalent:*

(i) *t refines s*

(ii) *for all ϕ of Hennessy-Milner logic, the judgment $\mathbb{V}(s, \phi)$ implies the judgment $\mathbb{V}(t, \phi)$*

(iii) *for all ϕ of Hennessy-Milner logic, the judgment $\mathbb{S}(t, \phi)$ implies the judg-*

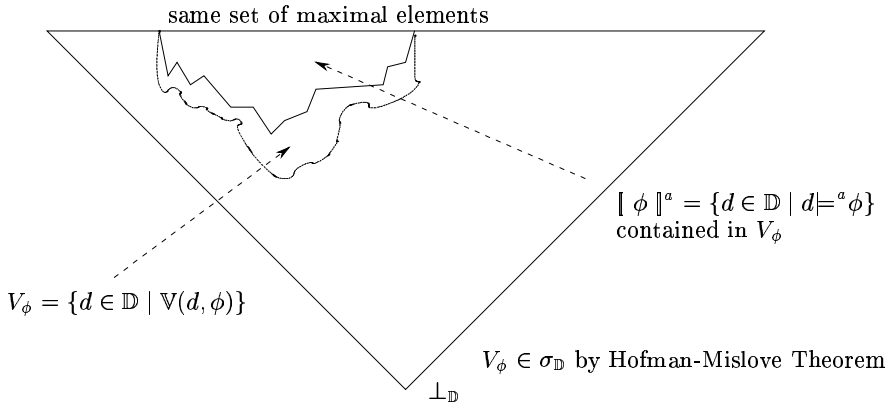

Fig. 8. Schematic of the potential loss of precision of the semantics based on $s \models^a \phi$ over the one based on validity judgments $\mathbb{V}(s, \phi)$.

ment $\mathbb{S}(s, \phi)$.

Items (ii) and (iii) are equivalent by duality. That (i) implies (ii) follows from (28) and item (ii) on page 7. To show that (ii) implies (i) we use the completeness of refinement and (28) again.

Example 6.3 Figure 8 shows a potential loss of precision of the judgment $t \models^a \phi$ over the validity judgment $\mathbb{V}(t, \phi)$. This loss of precision is captured in the domain $\mathbb{D}$ by the set $V_\phi \setminus [\![\phi]\!]^a$.

The completeness of refinement has a clear logical meaning, for we can show it to be equivalent to the fact that for all $k \in \mathbf{K}(\mathbb{D})$ no loss of precision occurs for ϕ_k: $[\![\phi_k]\!]^a = V_{\phi_k}$.

Theorem 6.4 *[[Hut04b]] Refinement of modal transition systems is complete for implementations iff for all $k \in \mathbf{K}(\mathbb{D})$ we have that the judgments $\mathbb{V}(s, \phi_k)$ and $s \models^a \phi_k$ are the same for all states s of all modal transition systems.*

6.4 Some research questions

Since the sets $[\![\phi]\!]^a$ are $\lambda_{\mathbb{D}}$-clopen for all ϕ of Hennessy-Miler logic, we can infer that $[\![\phi]\!]^a$ is the finite union of sets $[\![\phi_k]\!]^a$ and so, with respect to $\models^a$, each ϕ is the finite disjunction of Hennessy-Milner logic formulas for which validity checking on implementation classes is model checking by Theorem 6.4. This triggers several questions:

- For which additional ϕ of Hennessy-Milner logic is $V_\phi = [\![\phi]\!]^a$? For these ϕ, validity checking is reducible to model checking in constant time and space.
- For which ϕ of Hennessy-Milner logic is V_ϕ $\lambda_{\mathbb{D}}$-closed and therefore of the form $[\![\psi]\!]^a$ for some ψ of Hennessy-Milner logic? For these ϕ, validity checking is reducible to model checking but such reductions may have non-trivial complexities.

- What can we say about the topological complexities of the sets $\| \phi \|^a$ and V_ϕ if ϕ ranges over formulas of the modal mu-calculus [Koz83]?
- Finally, can one use Wadge reducibility [Wad83] and the theory of (reflective) φ-spaces [Sel04] to develop a descriptive set theory for sets of the previous item, and would such a description be related to expressiveness results of the modal mu-calculus hierarchy?

7 Conclusions

In this paper we presented a domain model $\mathbb{D}$ that forms the state space of a modal transition system $\mathcal{D}$ such that the latter is universal for all modal transition systems and the former is fully abstract, the domain order is the greatest refinement relation on $\mathcal{D}$. This was joint work with Radha Jagadeesan and David Schmidt and has been reported in [HJS04] already. Then we showed that the maximal points-space of $\mathbb{D}$ is a Stone space and indeed the quotient space of labelled transition systems modulo bisimulation, leading to robust consistency measures between modal transition systems and a compactness theorem for implementation classes of such systems over Hennessy-Milner logic. This work has also been reported elsewhere in some detail [Hut04a]. Finally we sketched the work in [Hut04b], that the greatest binary refinement relation (a co-inductive notion) is exactly reverse containment of classes of implementations, as one would hope for and expect intuitively. We pointed out that this completeness of refinement has an interesting characterization in terms of reducing validity or satisfiability checks to model checks [Hut04b] and leads to open problems in such reducibility questions for temporal logics in general. We refer to the papers [HJS01,HJS04,Hut04a,Hut04b,HH04] for a more complete account of related work and the results and proof outlines discussed or mentioned in this paper.

Acknowledgments

The technical material and results in Section 3 had been developed with Radha Jagadeesan and David Schmidt in [HJS01,HJS04]. We thank the organizers of the Third Irish Conference on the Mathematical Foundations of Computer Science and Information Technology for their kind invitation to give the talk on which this paper is based. Victor Selivanov kindly alerted us to related work on Wadge reducibility and φ-spaces [Sel04].

References

[Abr91] S. Abramsky. A domain equation for bisimulation. *Information and Computation*, 92(2):161–218, 1991.

[AJ94] S. Abramsky and A. Jung. Domain theory. In S. Abramsky, D. M. Gabbay, and T. S. E. Maibaum, editors, *Handbook of Logic in Computer Science*, volume 3, pages 1–168. Oxford University Press, 1994.

[BG99] G. Bruns and P. Godefroid. Model Checking Partial State Spaces with 3-Valued Temporal Logics. In *Proc. of the 11th Conference on Computer Aided Verification*, volume 1633 of *Lecture Notes in Computer Science*, pages 274–287. Springer Verlag, July 1999.

[BPR01] T. Ball, A. Podelski, and S. K. Rajamani. Boolean and Cartesian Abstraction for Model Checking C Programs. In T. Margaria and W. Yi, editors, *Proc. of TACAS'2001*, volume 2031 of *LNCS*, pages 268–283, Genova, Italy, April 2001. Springer Verlag.

[CC77] P. Cousot and R. Cousot. Abstract interpretation: a unified lattice model for static analysis of programs. In *Proc. of the 4th ACM Symp. on Principles of Programming Languages*, Los Angeles, California, 1977.

[DGG97] D. Dams, R. Gerth, and O. Grumberg. Abstract interpretation of reactive systems. *ACM Transactions on Programming Languages and Systems*, 19:253–291, 1997.

[Dam96] D. Dams. *Abstract interpretation and partition refinement for model checking.* PhD thesis, Technische Universiteit Eindhoven, The Netherlands, 1996.

[dBZ82] J. W. de Bakker and J. I. Zucker. Denotational Semantics Of Concurrency. In *Proc. of the 14th Annual ACM Symposium on Theory of Computing*, pages 153–158, New York, New York, 1982. ACM Press.

[EJS03] S. Eisenbach, V. Jurisic, and C. Sadler. Modeling the evolution of .NET programs. In *IFIP International Conference on Formal Methods for Open Distributed Systems*, Lecture Notes in Computer Science. Springer Verlag, 2003.

[GJ03] P. Godefroid and R. Jagadeesan. On The Expressiveness of 3-Valued Models. In L. D. Zuck, P. C. Attie, A. Cortesi, and S. Mukhopadhyay, editors, *Proc. of the 4th Conference on Verification, Model Checking and Abstract Interpretation*, volume 2575 of *Lecture Notes in Computer Science*, pages 206–222, New York, January 2003. Springer Verlag.

[Gun92] C. Gunter. The mixed power domain. *Theoretical Computer Science*, 103:311–334, 1992.

[Hec90] R. Heckmann. Set Domains. In *Proc. of the Third European Symposium on Programming*, pages 177–196, Copenhagen, Denmark, 1990. Springer Verlag.

[HH04] A. Hussain and M. Huth. On model checking multiple hybrid views. In T. Margaria, B. Steffen, A. Philippou, and M. Reitenspiess, editors, *Preliminary Proc. of the 1st International Symposium on*

Leveraging Applications of Formal Methods, Technical Report TR-2004-6, Department of Computer Science, University of Cyprus, pages 235–242, Paphos, Cyprus, 30 October - 2 November 2004.

[HJS01] M. Huth, R. Jagadeesan, and D. A. Schmidt. Modal transition systems: a foundation for three-valued program analysis. In Sands D., editor, *Proc. of the European Symposium on Programming*, volume 2028 of *Lecture Notes in Computer Science*, pages 155–169. Springer Verlag, 4-6 April 2001.

[HJS04] M. Huth, R. Jagadeesan, and D. Schmidt. A domain equation for refinement of partial systems. *Mathematical Structures in Computer Science*, 14(4):469–505, 5 August 2004.

[HM81] K. H. Hofmann and M. Mislove. Local compactness and continuous lattices. In B. Banaschewski and R.-E. Hoffmann, editors, *Continuous Lattices*, volume 871 of *Lecture Notes in Mathematics*, pages 209–248, Bremen, Germany, 1981. Springer Verlag.

[HM85] M. Hennessy and R. Milner. Algebraic laws for nondeterminism and concurrency. *Journal of the ACM*, 32(1):137–161, January 1985.

[Hut04a] M. Huth. Beyond image-finiteness: labelled transition systems as a Stone space. In *Proc. of the Nineteenth Annual IEEE Symposium on Logic in Computer Science*, pages 222–231, Turku, Finland, 13-17 July 2004. IEEE Computer Society Press.

[Hut04b] M. Huth. Refinement is complete for implementations. Submitted to an international journal, January 2004.

[Joh82] P. T. Johnstone. *Stone spaces.* Number 3 in Cambridge studies in advanced mathemathics. Cambridge University Press, 1982.

[Koz83] D. Kozen. Results on the propositional mu-calculus. *Theoretical Computer Science*, 27:333–354, 1983.

[KS90] P. C. Kanellakis and S. A. Smolka. CCS expressions, finite state processes, and three problems of equivalence. *Information and Computation*, 86(1):43–68, May 1990.

[Lar89] K. G. Larsen. Modal Specifications. In J. Sifakis, editor, *Automatic Verification Methods for Finite State Systems*, number 407 in Lecture Notes in Computer Science, pages 232–246, Grenoble, France, 12-14 June 1989. Springer Verlag.

[Law97] J. Lawson. Spaces of Maximal Points. *Mathematical Structures in Computer Science*, 7(5):543–555, 1997.

[LT88] K. G. Larsen and B. Thomsen. A Modal Process Logic. In *Proc. of the Third Annual IEEE Symposium on Logic in Computer Science*, pages 203–210. IEEE Computer Society Press, 5-8 July 1988.

[Plo81] G. D. Plotkin. A Structural Approach to Operational Semantics. Technical Report FN - 19, DAIMI, Copenhagen, Denmark, September 1981.

[Sco72] D. S. Scott. *Formal semantics of programming languages*, volume 2 of *Courant Computer Science Symposia*, chapter "Lattice theory, data types and semantics," pages 65–106. Prentice-Hall, 1972.

[Sel04] V. Selivanov. Difference hierarchy in φ-spaces. *Algebra and Logic*, 43(4):425–444, 2004. In Russian, English translation available.

[Sti96] C. Stirling. Games and Modal Mu-Calculus. In T. Margaria and B. Steffen, editors, *Proc. of the Second International Workshop in Tools and Algorithms for Construction and Analysis of Systems*, volume 1055 of *Lecture Notes in Computer Science*, pages 298–312, Passau, Germany, 27-29 March 1996. Springer Verlag.

[Wad83] W. Wadge. *Reducibility and determinateness in the Baire space*. PhD thesis, University of Berkeley, Berkeley, California, 1983.

A Basic terminology and notation from domain theory

(We recommend [AJ94] for a thorough reference on these issues.) A partial order $(D, \leq)$ is a set D with a binary relation $\leq$ on D that is reflexive, transitive, and antisymmetric. An upper bound for a subset A of a partial order D is an element $u \in D$ such that $a \leq u$ for all $a \in A$; we write $ub(A)$ for the set of upper bounds of A. The set $mub(A) = \{u \in ub(A) \mid \forall d \in D\colon d \leq u \;\&\; d \in ub(A) \Rightarrow d = a\}$ consists of all minimal upper bounds of A in D.

A subset A of D is directed iff all finite subsets of A have an upper bound in A. Given $X \subseteq D$ we write $\downarrow_D X$ for $\{d \in D \mid \exists x \in X\colon d \leq x\}$, $\uparrow_D X$ for $\{d \in D \mid \exists x \in X\colon x \leq d\}$, and elide the subscript D if it is determined by context. We use $\downarrow x$ and $\uparrow x$ if $X = \{x\}$. Subsets U of D with $U = \uparrow U$ are upper sets, and subsets L of D with $L = \downarrow L$ are lower sets.

A partial order $(D, \leq)$ is a dcpo iff all its directed subsets have a least upper bound $\bigvee A$, i.e. iff there is some $\bigvee A \in D$ with $ub(A) = \uparrow \bigvee A$. We write $\bot_D$ for the unique element of D satisfying $\bot_D \leq d$ for all $d \in D$, provided such an element exists. An element $k \in D$ is compact in a dcpo D iff for all directed sets A of D with $k \leq \bigvee A$ there is some $a \in A$ with $k \leq a$; we write $\mathbf{K}(D)$ for the set of compact elements. A dcpo D is algebraic iff for all $d \in D$ the set $\{k \in \mathbf{K}(D) \mid k \leq d\}$ is directed with least upper bound d; if in addition $\mathbf{K}(D)$ is countable, then D is ω-algebraic. For a finite subset F of D define $mub^1(F) = mub(F)$, $mub^{n+1}(F) = mub(mub^n(F))$ for all $n \geq 1$, and $mub^\infty(F) = \bigcup_{n \geq 1} mub^n(F)$. A bifinite domain is an algebraic dcpo D such that for every finite subset $F \subseteq \mathbf{K}(D)$ the set $mub^\infty(F)$ is finite and contained in $\mathbf{K}(D)$ with $ub(F) = \uparrow mub(F)$. The class of ω-algebraic bifinite domains is closed under finite products $\prod_{i=1}^n D_i$ in their point-wise order and $\mathbf{K}(\prod_{i=1}^n D_i) = \prod_{i=1}^n \mathbf{K}(D_i)$.

A topological space (X, τ) consists of a set X and a family τ of subsets of X such that $\{\}$ and X are in τ, and τ is closed under finite intersections and arbitrary unions. Elements $O \in \tau$ are τ-open, complements $X \setminus O$ with $O \in \tau$ are τ-closed, and sets that are τ-open and τ-closed are τ-clopen. A topological space (X, τ) is τ-compact iff for all $\mathcal{U} \subseteq \tau$ with $X \subseteq \bigcup \mathcal{U}$ there is a finite subset $\mathcal{F} \subseteq \mathcal{U}$ with $X \subseteq \bigcup \mathcal{F}$. A topological space (X, τ) is T_0 iff for all $x \neq y$ in X there is some $O \in \tau$ that contains exactly one of x and y. A subset A of X is dense in (X, τ) iff $A \cap O$ is non-empty for all non-empty $O \in \tau$.

Given a topological space (X, τ) and a subset $Y \subseteq X$, the subspace topology on Y consists of the set $\{O \cap Y \mid O \in \tau\}$. A subset Y of X is τ-compact iff Y is compact in its subspace topology. A subset Y is τ-saturated in X iff Y is the intersection of τ-open sets. A metric on X is a function $d\colon X \times X \to [0, 1]$ such that for all $x, y, z \in X$ we have

(i) $d(x, y) = 0$ iff $x = y$

(ii) $d(x, y) = d(y, x)$ and

(iii) $d(x, z) \leq d(x, y) + d(y, z)$.

A metric $d\colon X \times X \to [0, 1]$ incudes a topology τ_d on X whose elements are all those $O \subseteq X$ that are unions of sets of the form $B_\eta(x) = \{y \in X \mid d(x, y) < \eta\}$ for $x \in X$ and rational $\eta > 0$. A topological space (X, τ) is metrizable iff there is a metric $d\colon X \times X \to [0, 1]$ such that $\tau = \tau_d$.

The definitions and characterizations below assume that D is a bifinite domain. The Scott-topology on D consists of all subsets U of D satisfying $U = \uparrow(U \cap \mathbf{K}(D))$; such elements are Scott-open. The Lawson-topology on D consists of all subsets V of D such that $x \in V$ implies the existence of some $k, l \in \mathbf{K}(D)$ with $x \in \uparrow k \setminus \uparrow l \subseteq V$. Note that every Scott-open is Lawson-open and every Scott-closed is therefore Lawson-closed. For all $d \in D$, the set $\uparrow d$ is Lawson-closed upper. A subset C of D is Scott-compact (Scott-)saturated in D iff C is Lawson-closed upper in D. A subset U of D is Scott-open and Scott-compact iff U is of the form $\uparrow F$ for a finite set $F \subseteq \mathbf{K}(D)$.

A collection $(F_i)_{i \in I}$ of subsets of D, indexed by a directed set $(I, \leq)$, is filtered iff (for all $i, j \in I$ there is some $k \in I$ with $k \in ub(\{i, j\})$ such that $F_k \subseteq F_i \cap F_j$). The Hofmann-Mislove Theorem [HM81] states that if the intersection $\bigcap_{i \in I} C_i$ of a filtered collection of Scott-compact saturated sets $(C_i)_{i \in I}$ in D is contained in a Scott-open set $U \subseteq D$, then there is some $i_0 \in I$ with $C_{i_0} \subseteq U$ already.