Johan van Benthem

Program Constructions that are Safe for Bisimulation*

Abstract It has been known since the seventies that the formulas of modal logic are invariant for bisimulations between possible worlds models — while conversely, all bisimulation-invariant first-order formulas are modally definable. In this paper, we extend this semantic style of analysis from modal formulas to dynamic program operations. We show that the usual regular operations are *safe for bisimulation*, in the sense that the transition relations of their values respect any given bisimulation for their arguments. Our main result is a complete syntactic characterization of all first-order definable program operations that are safe for bisimulation. This is a semantic functional completeness result for programming, which may be contrasted with the more usual analysis in terms of computational power. The 'Safety Theorem' can be modulated in several ways. We conclude with a list of variants, extensions, and further developments.

1. Bisimulation in Modal and Dynamic Logic

In current mathematical theories of computation, 'bisimulation' between labeled transition systems has become a natural measure for equivalence of processes. Essentially, a *labeled transition system* is a set of states with a family of binary transition relations over these:

$$(S, \{R_a\}_{a \in A}),$$

where certain unary predicates over states may also be present (common examples are 'success' or 'dead-lock'). Now, the basic process equivalence is as follows (cf. Park 1981):

Definition. A *bisimulation* between two labeled transition systems is a binary relation C between their state sets satisfying 'atomic harmony' as well as two zigzag clauses allowing 'mutual tracing' of the process:

(i) if $s\,C\,s'$, then s, s' validate the same atomic propositions,

(ii) if $s\,C\,s'$ and $s\,R_a t$, then there is some t' with $s' R_a t'$ and $t\,C\,t'$; and vice versa.

A 'computational process' may now be viewed as a class of labeled transition systems that is closed under bisimulation. The same notions had al-

[*] A version of this paper was presented at Logic Colloquium Conference held in Clermont-Ferrand, July 21–30, 1994.

Presented by **Denis Richard**, **Jerzy Tomasik** and **Jan Zygmunt**; *Received* December 1, 1995; *Revised* June 16, 1997

Studia Logica **60**: 311–330, 1998.

ready emerged in Modal Logic (cf. van Benthem 1976). Bisimulation is the key semantic invariance for the modal language describing labeled transition systems, viewed as poly-modal Kripke models, which has the usual Boolean operators as well as indexed modalities $\langle a \rangle$ for each atomic action $a \in A$. Modal formulas are *invariant for bisimulation*, in the following sense. Whenever C is a bisimulation between two models M, M' with $s\,C\,s'$, we have

$$M, s \models \varphi \quad \text{iff} \quad M', s' \models \varphi \qquad\qquad \text{for all modal formulas } \varphi.$$

Atomic harmony (i) provides the induction base here, and the back-and-forth clauses (ii) are just what is needed here to push the induction through for the existential modalities. This observation can also be reversed, showing that the above 'bisimulation invariance' is indeed the defining semantic characteristic of the modal formalism. To state the relevant model-theoretic preservation result, one views the latter formalism as a fragment of the appropriate first-order description language for Kripke models (using the well-known 'standard translation', cf. van Benthem 1984). Then, we have the following result:

THEOREM. *A first-order formula $\varphi(x)$ over labeled transition systems is invariant for bisimulation if and only if it is definable by means of a modal formula.*

The original proof of this Invariance Theorem is in van Benthem 1976, a short general version using ω-saturated models (cf. Chang & Keisler 1973) is in van Benthem 1991A.

This style of analysis is preserved when passing from basic Modal Logic to Propositional Dynamic Logic (Harel 1984), which also descibes complex transitions (induced by compound 'programs') over labeled transition systems by means of regular program expressions π, including tests, and their corresponding modalities $\langle \pi \rangle$. Here again, there is invariance of dynamic formulas φ for bisimulations C between two models — but there is also a new aspect. Intertwined with the old proof, one also has to show that the usual back-and-forth clauses in bisimulation are inherited by the regular program constructions. Indeed, each binary transition relation $[\![\pi]\!]$ shows this behaviour, upward from the atomic ones. More precisely, a joint induction on programs and formulas shows:

PROPOSITION. *If C is a bisimulation between two models M, M', with $s\,C\,s'$, then*

(i) *s, s' verify the same formulas of propositional dynamic logic,*

(ii) *whenever $s\,[\![\pi]\!]^{M} t$, then there exists t' with $s'\,[\![\pi]\!]^{M'} t'$ and $s'\,C\,t'$.*

This observation motivates the following notion of invariance for program operations (where we indulge in a slight abuse of notation, for greater readability):

DEFINITION. An operation $O(R_1, \ldots, R_n)$ on programs is *safe for bisimulation* if, whenever C is a relation of bisimulation between two models for their transition relations $R_1, \ldots, R_n$, then it is also a bisimulation for the defined relation $O(R_1, \ldots, R_n)$.

Also independently from the preceding modal analysis, safety for bisimulation seems to be an interesting general semantic criterion for admissible basic programming operations. It is easy to show that, e.g., the regular operations of relational composition ; and choice $\cup$ (i.e., Boolean union) have this property. Another example are the standard test relations $(\varphi)?$ for modal formulas φ. All these examples reflect the key observations in the straightforward inductive proof of the previous Proposition. For later reference, we also mention the safety of one less familiar negation operation, widely employed in the recent literature on so-called 'dynamic semantics' (cf. van Benthem 1996):

$$\sim(R) \;=\; \{(x,y) \mid x = y \text{ and for no } z : xRz\} \qquad \text{'counter-domain'}$$

Now, the following natural question arises:

> *Is there some companion to the earlier preservation theorem for modal invariance, characterizing those programming operations that guarantee safety for bisimulation?*

More specifically, one may ask whether the semantic criterion of safety for bisimulation gives us precisely the *regular* programming operations that have been so prominent for independent computational reasons. Again, to make the question precise, we go to the standard first-order description language over labeled transition systems — this time, adding arbitrary binary relation symbols for transition relations. Let us call a programming operation *first-order* if can be defined using a formula $\theta(x, y)$ of this language with two free variables. All earlier operations are first-order in this sense, witness their definitions:

$(R_1 ; R_2)$	$\exists z\,(R_1xz \wedge R_2zy)$
$(R_1 \cup R_2)$	$R_1xy \vee R_2xy$
$\sim(R)$	$x = y \wedge \neg\exists z\, Rxz$
$(P)?$	$x = y \wedge Px$

The main result of this paper is the following complete semantic characterization:

THEOREM. *A first-order relational operation $O(R_1, \ldots, R_n)$ is safe for bisimulation iff it can be defined using atomic relations $R_a xy$ and atomic tests $(q)?$ for propositional atoms q in our models, using the three operations* $;$, $\sim$ *and* $\cup$.

The proof of the Safety Theorem is somewhat complex — whence we postpone the main argument until Section 3 below. It turns on a Lemma which expresses a model-theoretic fact about Modal Logic, namely, a preservation theorem for 'continuous' modal formulas. We prove the latter result separately in Section 2, for its independent interest.

2. Continuous modal formulas

Let us call a modal formula $\varphi(p)$ *continuous* in the proposition letter p if (with some abuse of notation), the following equivalence holds in each model:

For each family of subsets $\{P_i\}_{i\in I}$, $\varphi(\bigcup_{i\in I} P_i) \leftrightarrow \bigvee_{i\in I} \varphi(P_i)$.

Examples are $p \wedge q$, $\langle a\rangle p \wedge \langle b\rangle \neg q$, $p \vee \langle a\rangle p$, and non-examples are $\neg p$, $[a]p$. We seek a syntactic characterization for this notion. Some precedents exist. Continuity (properly) implies the well-known property of semantic *monotonicity*, whose syntactic correlate is definability of φ using only *positive occurrences* for the proposition letter p. (For first-order predicate logic, this is the well-known Lyndon Theorem, which also holds for basic modal logic. Cf. Andréka, van Benthem & Németi 1995.) Thus, we expect some even stricter syntactic constraint here, which is provided by the following result:

THEOREM. *Modulo logical equivalence, the p-continuous modal formulas $\varphi(p)$ are all those that can be written as disjunctions of formulas of the 'existential forms' $\alpha_0 \wedge p$, $\alpha_0 \wedge \langle a_1\rangle(\alpha_1 \wedge p)$, $\alpha_0 \wedge \langle a_1\rangle(\alpha_1 \wedge \langle a_2\rangle(\alpha_2 \wedge p))$, etcetera, where all formulas α_i are p-free.*

Indeed, for standard predicate logic, a somewhat similar syntactic characterization of Continuity is not hard to find (cf. van Benthem 1986), but the straightforward argument given there cannot be reproduced within the modal fragment.

PROOF. In one direction, it is easy to check that all forms described are indeed continuous with respect to the proposition letter p. The hard part, as usual, is the converse, which we approach via the following auxiliary assertion:

CLAIM. A continuous formula φ implies the infinite disjunction of all those modal existential forms as described above which themselves imply φ as a consequence.

Then, by Compactness, φ will imply some finite disjunction of these forms, and hence it will be equivalent to the latter, since it follows from each disjunct. Thus, it remains to prove the Claim (which is where the real work lies).

Let $M, w \vDash \varphi$. By Continuity in its downward direction, using the fact that the set $\mathrm{V}(p)$ is the union of its singletons, we have $M', w \vDash \varphi$, where p holds at only one world. (Note that $\mathrm{V}(p)$ can never be empty: since $\varphi(\emptyset)$ would imply an empty disjunction, which is a contradiction.) We may assume that M', w is generated from w (since φ is modal, this makes no difference in truth value). Thus, let there be a finite path with labeled transitions $w = w_0\, R_1\, w_1\, R_2 \ldots R_n\, w_n \vDash p$ from the root to the unique world where p holds. Let Φ_i be the set of p-free modal formulas that are true at w_i. Via the usual first-order translation, one can also think of all modal formulas here as first-order ones. Now, the following valid semantic consequence holds in first-order logic:

SUBCLAIM. $\Phi_0(x_0), R_1x_0x_1, \Phi_1(x_1), \ldots, R_nx_{n-1}x_n, \Phi_n(x_n), Px_n \vDash \varphi.$

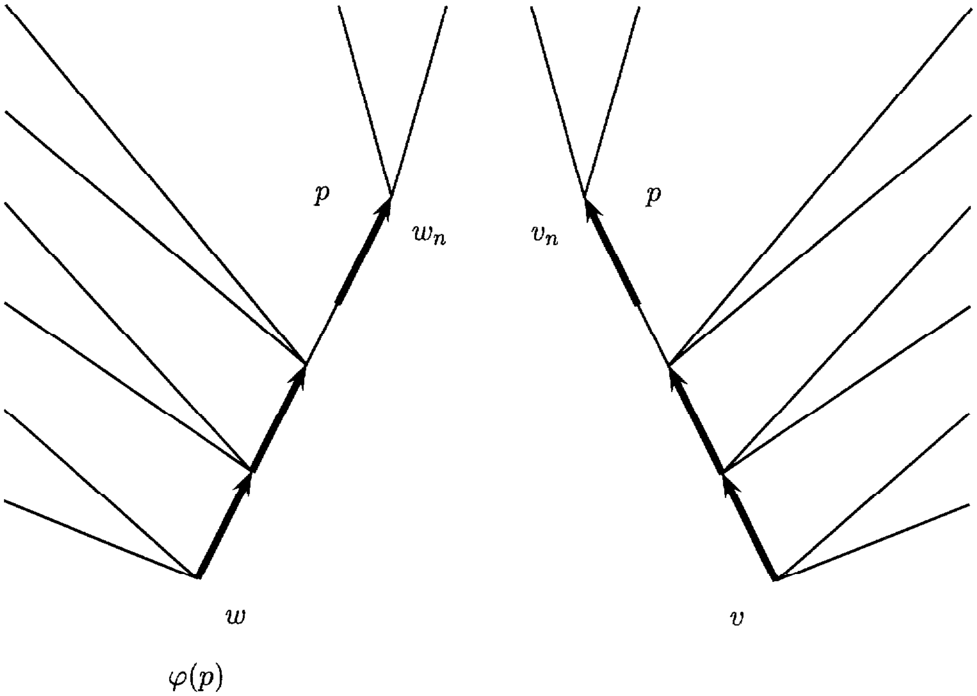
If we can show this, then once more by Compactness, a sequence of finite subsets of the Φ_i will do in the premise — which yields the required existential form implying φ by some straightforward predicate-logical manipulation. Generalizing over all models, then, we have shown that the formula φ locally implies some such existential form everywhere: and hence it implies their infinite disjunction globally.

PROOF OF THE SUBCLAIM. Consider any model N for the set of formulas $\Phi_0(x_0)$, $R_1x_0x_1$, $\Phi_1(x_1)$, $\ldots$, $R_nx_{n-1}x_n$, $\Phi_n(x_n)$, with the required n-sequence $v_0, v_1, \ldots, v_n$ for its free variables starting at v. As in modern proofs of the preservation theorem for modal formulas with respect to bisimulation (cf. van Benthem 1991A), we may suppose without loss of generality that

(i) M', N are *ω-saturated*
(we are only dealing with first-order formulas);

(ii) M', N are *intransitive trees* via some 'unraveling bisimulation':
(we are in fact only dealing with modal formulas).

We can draw these models as indicated, each with a distinguished branch of length $n+1$, and the rest of the models lying in disjoint subtrees branching off from these:



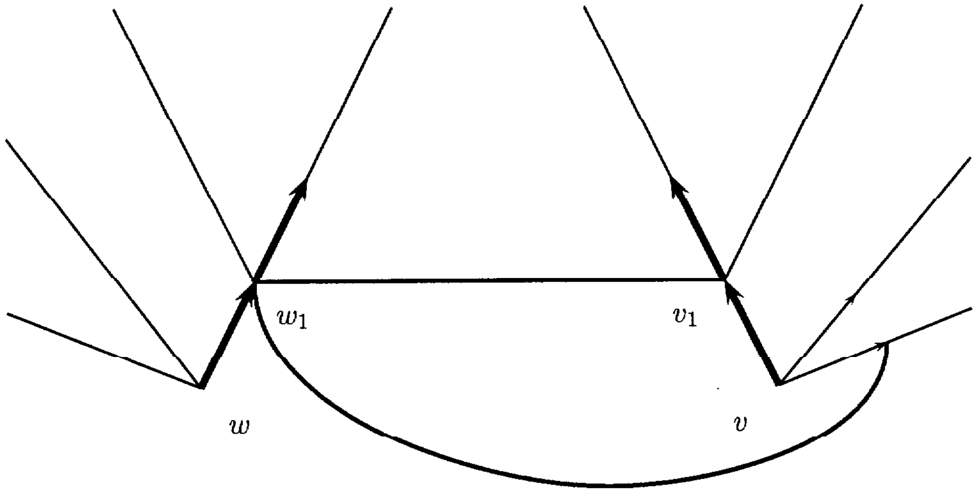
Moreover, the following stipulation between states:

$$x \equiv u \quad \text{iff} \quad M', x \text{ and } N, u \text{ verify the same } L\text{-modal formulas}$$

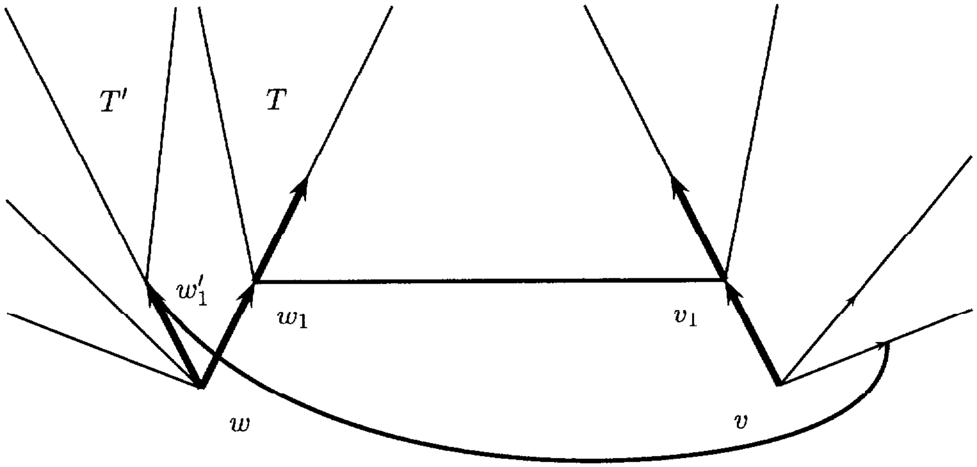
defines an L-bisimulation between these two models. This may be shown by the usual zigzag argument on saturated models — as in the proof of the modal Invariance Theorem. In particular, this stipulation matches corresponding points on the above two special branches, by the definition of the sets Φ_i. Now, our purpose is to further improve this L-bisimulation to a $(L+P)$-bisimulation respecting also the propositional atom p, so that we can transfer the truth of the initial formula $\varphi(p)$ from left to right. For this purpose, judicious geometrical rearrangement will be used on labeled transition graphs, to produce a situation where the matching on the two branches is unique (in particular, the final world w_n corresponds only to v_n), while subtrees on the left only contain matches in their corresponding subtrees on the right — and vice versa.

For a start, without loss of generality, we may assume that all bisimulation links occur between worlds at the same levels in the trees (any others may be omitted without losing the bisimulation property). Here are the main steps in our procedure, which works upward along the special branches. As often in possible worlds semantics, it helps to visualize what is going on using suitable graph pictures:

(1a) Start with the match w_1, v_1. Suppose that world w_1 has links with other level-one worlds in N, as shown in the following picture:

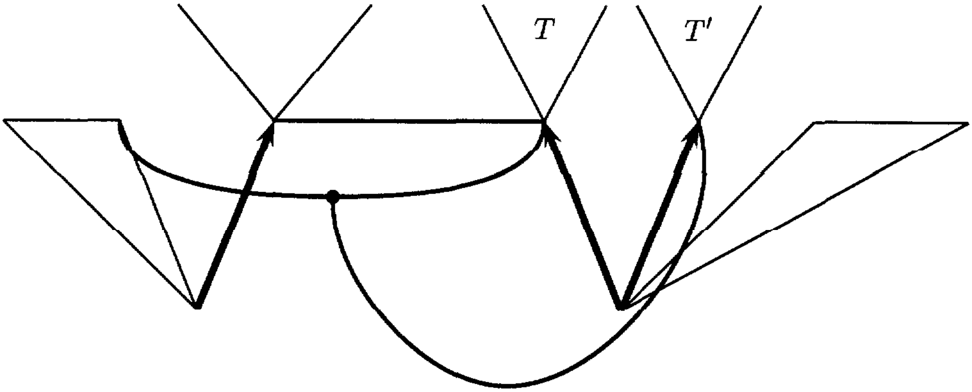


Then 'de-couple' by making a copy of the subtree T, w_1 on the left, attaching this to the root w, and matching the former mates of w_1 on the right to its copied companion w_1' (and so on upward in the graph for the relevant relational successors). This transformation is pictured below:



The result is still an L-bisimulation between the model on the right and the new enlarged one on the left, while the latter also L-bisimulates with the original M. Now, in the enlarged model, a duplication occurs of the single world w_n with p, which is not what we want. But applying the downward half of continuity again, we see that $\varphi(p)$ will still hold at w if we have p true at just one of its two locations. Moreover, by the isomorphism of copies, we may suppose that this happens at the original location, without loss of generality.

(1b) Likewise, matches between v_1 and sisters of w_1 can be decoupled as shown in the following picture, which again involves copying, but now without any special moves for economizing on p-worlds:



(2) The effect of the previous two moves is that the initial match w_1, v_1 has become unique — and we can now repeat the procedure, moving upward along the special branches until we reach the final match w_n, v_n, and make that unique too.

Now, let M^* be the enlarged model on the left-hand side (which has p true only at the end of the special branch), and N^* its companion on the right-hand side with p only true at the end of its special branch. The original enlarged N^+ is this same model, but with possibly more occurrences of p in other worlds. By the construction, the model N^* $(L{+}P)$-bisimulates with M, such that v matches in both models. Then, our final argument traces the truth of $\varphi(p)$ in these successive models as follows:

- $\varphi(p)$ still holds at w in the enlarged model M^* (because of the construction),
- via the $(L{+}P)$-bisimulation between M^* and N^*, it will transfer to v in N^*,

- by upward monotonicity [the other side of continuity, which had not been used so far in this proof], $\varphi(p)$ will also hold at v in N^+,
- by $(L{+}P)$-bisimulation, it will also hold at v in N: which was to be shown. ■

3. Proof of the Safety Theorem

Having obtained an explicit syntactic characterization of Continuity, we are ready to give the proof for our main result. It may be broken up into the following steps.

The given format is safe

(1) By our earlier observations, each operation of the described kind is safe for bisimulation. Here are two examples. *Composition.* Let $x\,R\,;S\,y$ in M_1, and $x\,C\,u$, with u in M_2. There exists z in M_1 with $x\,R\,z$ and $z\,S\,y$. By bisimulation for R, one finds a v in M_2 such that $u\,R\,v$ and $z\,C\,v$. Then, by bisimulation for S, one finds a w in M_2 such that $v\,S\,w$ and $y\,C\,w$. The latter is the required $R;S$-successor of u in M_2. *Counterdomain.* Let $x \sim\!(R)\,y$ in M_1: that is, $x = y$ and x has no R-successors. Let $x\,C\,u$ in M_2. Now, suppose that u had some R-successor in M_2. By bisimulation for R, there would also be a corresponding R-successor for x in M_1: quod non. Thus, (u, u) is a matching $\sim\!(R)$-successor step in M_2.

Safety Stays Inside The Format

(2) Next, let $\theta(x, y)$ be a first-order operation, in a language L, which is safe for bisimulation. Note that invariance for bisimulation is really a language-dependent notion: what matters is if the bisimulation goes back-and-forth with respect to the atomic binary relations, and whether its matchings of states respect the relevant unary predicates. Next, choose a *new* unary predicate letter P.

CLAIM. The formula $\exists y\,(\theta(x, y) \wedge Py)$ is invariant for $(L{+}P)$-bisimulation.

PROOF. Immediate from the safety of θ for L-bisimulation, plus the obvious fact that $(L{+}P)$-bisimulations are also L-bisimulations. ■

(3) By the earlier-mentioned characterization of modal formulas, this means that the first-order formula $\exists y\,(\theta(x, y) \wedge Py)$ must be equivalent to some *modal formula* $\varphi(p)$. Moreover, given the shape of our first-order formula,

this $\varphi(p)$ will be *continuous* in the proposition letter p. But then, it will be definable using a disjunction of formulas

$$\alpha_0 \wedge \langle a_1\rangle(\alpha_1 \wedge \ldots \wedge \langle a_n\rangle(\alpha_n \wedge p)\ldots),$$

where the α_i do not contain the proposition letter p.

Now, it is straightforward to check the following assertion:

CLAIM. θ will be definable using the corresponding union of relations of the form

$$(\alpha_0)?\,;a_1\,;(\alpha_1)?\,;\ \ldots\ ;a_n\,;(\alpha_n)?$$

(4) Finally, one can remove possibly complex modal tests in the latter schema:

CLAIM. All complex modal tests $(\varphi)?$ can be reduced to atomic ones using only the regular operations ; and $\sim$.

PROOF. This may be done using the following three valid identities to push all tests inward:

$$(\varphi \wedge \psi)? = (\varphi)?\,;(\psi)? \quad (\neg\varphi)? = {\sim}(\varphi)? \quad (\langle a\rangle\varphi)? = {\sim}{\sim}(a\,;(\varphi)?) \quad \blacksquare$$

4. Variations and extensions

4.1. Variants of the proof

The modal Safety Theorem was first published in a preprint four years ago (van Benthem 1993B). In the meantime, the above laborious proof has been simplified considerably by Marco Hollenberg (cf. Hollenberg 1997). The key proof step for the preservation theorem about continuous modal formulas $\varphi(p)$ compared two models with distinguished paths leading up to some p-world, that were created via successive model transformations (including repeated appeals to Compactness, modal tree unraveling, and 'ω-copying'). But in fact, so-called '2-unravelings' suffice, that is, standard tree unravelings with only two copies of each former node. Also, one merely needs to maintain the crucial modal equivalence between the successive nodes on the two distinguished paths in a sequence of decreasing syntactic complexity. (At the root, one requires equivalence for all modal formulas up to the modal operator depth of φ — afterwards, moving up the branches, one 'counts down' to at most operator depth zero.) The resulting argument can be recast without Compactness, appealing only to finite *modal Ehrenfeucht games*. In this manner, Hollenberg has specialized the modal Safety Theorem to the realm of *finite models* — following the constructive game methods of Rosen 1995, who first proved the natural 'finite model version' of the original modal Invariance Theorem.

The Safety Theorem evidently generalizes the modal Invariance Theorem, as invariant formulas correspond uniquely to safe test programs. Starting from this observation, Hollenberg 1996 derives a more general preservation result for joint invariance and safety, for first-order relational operations involving arbitrary finite arities. The resulting format is a highly expressive modal first-order language for specifying program operations.

4.2. Infinitary versions

The above analysis was restricted to programming operations definable in a first-order formalism. Nevertheless, there are many useful programming constructions that involve *infinitary operations*, usually definable in the logic $L_{\infty\omega}$ which extends first-order logic with infinite unions. Examples are regular Kleene Iteration, but also general Fixed-Point Recursion $\mu p \bullet \varphi(p)$ for formulas φ that are monotone in the proposition p. (Such fixed points always have explicit solutions by infinite disjunctive 'unwinding'.) The methods of previous Sections may be modified to cover this case to some extent. We give a sketch.

THEOREM. *The $L_{\infty\omega}$-formulas that are invariant for bisimulation are precisely the infinitary modal ones, constructed using arbitrary conjunctions and disjunctions.*

PROOF. We adapt the proof for the Invariance Theorem in van Benthem & Bergstra 1995 for the infinitary language $L_{\omega_1\omega}$ — which avoids compactness or saturation. Of course, infinitary modal formulas (defined in the obvious way) are all invariant for bisimulation. Now, consider the converse. Let $\varphi(x)$ be a formula of $L_{\infty\omega}$ without a modal equivalent. We construct two models M, N with a bisimulation E between them which has a link $a\,E\,b$ such that $M \models \varphi(a)$, $N \models \neg\varphi(b)$. This will refute invariance for bisimulation. These models will be constructed from *good triples* A, Σ, Δ where A describes the bisimulation, Σ the model M, and Δ the model N. But first, some auxiliary definitions.

(i) By *extended modal formulas* over a set of variables X we mean all formulas of $L_{\infty\omega}$ that are constructed using unary atoms Px $(x \in X)$, Boolean operations (finite or infinite), and existential modal quantifiers $\exists y\ (Rxy \wedge [y/u]\psi)$ (where $x, u \in X$). (One can find a more perspicuous normal form for such formulas, but we shall not need this.) If $|X| = 1$, these are just ordinary infinitary modal formulas. (ii) For convenience, we assume that all formulas have negations pushed inside to atoms, leaving only operators $\wedge$, $\vee$, $\forall$ $\exists$. (iii) Let $\mu = \max(\aleph_0, |\mathrm{TC}(\neg\varphi)|)$, where $\mathrm{TC}(\neg\varphi)$ is the transitive closure of the formula $\neg\varphi$, which contains (amongst others) all its infinitary subformulas. (iv) Choose two disjoint sets C, D of *new* individual constants of cardinality μ^+ (a regular cardinal). Henceforth, all *formulas* will be C- or D-substitution instances of subformulas of $\neg\varphi$. The total cardinality of this set is again μ^+. (In counting this size, recall that formulas of $L_{\infty\omega}$ have only finitely many free variables.) (v) Next, we call two sets of formulas Σ, Δ *modally inseparable* with respect to A if there is no extended modal formula α over a set of variables X (of cardinality smaller than μ^+) with $\Sigma \models \alpha(\boldsymbol{c})$ and $\Delta \models \neg\alpha(\boldsymbol{d})$ — for subsets $\boldsymbol{c}$, $\boldsymbol{d}$ of C, D where corresponding pairs c/x, d/x have the atom $c\,E\,d$ in A. (vi) A *good triple* A, Σ, Δ satisfies the following conditions: all three sets have cardinality $< \mu^+$, A consists of bisimulation atoms $c\,E\,d$, Σ consists of formulas with constants only from C, and likewise for Δ and D — while Σ, Δ are modally inseparable with respect to A. The good triples form a set.

Next,we state some closure conditions on good triples. The first of these are familiar from the use of infinitary 'consistency properties' for describing consistent diagrams of models (cf. Keisler 1971), whence we omit their detailed proofs. Let A, Σ, Δ be a good triple.

- adding to Σ all conjuncts of an infinitary conjunction in Σ again gives a good triple, and the same holds for infinitary conjunctions in Δ. (This addition does not affect modal separation, and it keeps the cardinality of Σ below μ^+.)
- adding to Σ all substitution instances of a universally quantified formula in Σ, with respect to all constants from C already occurring in Σ, again gives a good triple, and the same holds for universal quantifiers in Δ.
- each infinitary disjunction in Σ has at least one disjunct that can be added to Σ to produce a good triple, and the same holds for Δ. (If each disjunct leads to a modal separation for the extended triple, their disjunction will separate the original triple. Here, we need extended modal formulas, as the constants involved may be different. Also, some care is needed by choosing disjoint sets of variables for each disjunct.)
- each existentially quantified formula in Σ has a substitution instance with some constant c that is new to Σ and A which can be added to Σ to form a good triple, and the same holds for Δ. (Notice that the new constant cannot trigger new modal separations, since it has no available bisimulation atoms.)

The next requirements build in atomic harmony and zigzags for the bisimulation E:

- if $cE,d \in A$ and $Pc \in \Sigma$, then $A, \Sigma, \Delta \cup \{Pd\}$ is good. (If modal α separates Σ and $\Delta \cup \{Pd\}$, then $Px \wedge \alpha$ separates the original Σ, Δ.) And vice versa for Δ.
- if $cEd \in A$ and $Rcc' \in \Sigma$, there is an atom d' new to A, Σ, Δ such that $A \cup \{c'Ed'\}, \Sigma, \Delta \cup \{Rdd'\}$ is a good triple — and vice versa. (If some modal α separates Σ and $\Delta \cup \{Rdd'\}$, then $\Diamond_x\alpha$ separates the original Σ, Δ.)

Finally, we construct our models. We enumerate all 'tasks' at hand, with 'fair scheduling'. Tasks have the form of either C-(D-)formulas, to be verified in the model M (N), or bisimulation atoms plus R-successor atoms, to be matched in the opposite model. The total number of these tasks is μ^+ (by a simple cardinality calculation) — and hence we can enumerate them in a sequence T_α ($\alpha < \mu^+$), so that each task occurs cofinally often. Now, we construct a corresponding sequence of good triples, starting with an initial triple

$$\{cEd\}, \{\varphi(c)\}, \{\neg\varphi(d)\}.$$

The latter is good, as modal separation would imply modal definability for φ: *quod non.* Now, at each stage α, we take the component-wise union of all previous triples, and add formulas according to the scheduled task (if relevant), via the above closure properties. The final result is again a triple A^*, Σ^*, Δ^*. This defines two models M, N (over domains consisting of those constants from C, D, respectively, which occur in Σ^*, Δ^*) plus a binary relation E between them in the obvious way. An easy induction shows that

- C-formulas are in Σ iff they are true in M, and likewise for D-formulas and N.
- E is a bisimulation between M and N. ■

This is a heavy-duty argument, which becomes quite cumbersome when spelt-out in full formal detail (cf. van Benthem 1997A for some spin-off, though). A more elegant proof of the infinitary modal Invariance Theorem is given in Barwise & van Benthem 1996, using a Lindström-type argument crucially involving the so-called Boundednesss Theorem for $L_{\infty\omega}$. The latter replaces the appeals to Compactness in the standard proofs. (Interestingly, this analysis naturally yields interpolation theorems, too — and modal invariance becomes a special case of generalized 'consequence under bisimulation'.) In particular, van Benthem 1997B provides an infinitary generalization of our main Safety Theorem:

THEOREM. *A relational operation $O(R_1, \ldots, R_n)$ which is definable in $L_{\infty\omega}$ is safe for bisimulation if it can be defined from atomic relations $R_a xy$ and atomic tests $(q)?$ for propositional atoms q in our models, using only the three operations $;$, $\bigcup$ and $\sim$, where the unions may now be* infinitary.

Invariance and Safety theorems also make sense for the countably infinitary language $L_{\omega_1\omega}$ (cf. Keisler 1971, van Benthem 1991A, van Benthem & Bergstra 1995). But such results would still leave a substantial question for infinitary programming constructions. What is the semantic extra of the *regular* operations, over their safety for bisimulation? The correct view here may be that safety gives a *semantic* space of reasonable program operators, where infinite union is natural — whereas further restrictions to μ-calculus or regular programs suggest a differently motivated subhierarchy inside this semantic space, motivated by additional considerations of computational *complexity*.

Another way to go has been suggested by Gerard Renardel (p.c.). Restrict attention to some suitable *effective fragment* of $L_{\omega_1\omega}$, and then characterize the regular operations as the safe ones definable inside that fragment. An alternative take on 'computability' would restrict infinitary program operations to those that can be defined explicitly via *fixed-point operators*. An elegant illustration of this strategy is the so-called modal 'μ-calculus', i.e., modal logic with the above-mentioned unary fixed-point operators. A modal Invariance Theorem was proved in Janin & Walukiewicz 1996, stating that formulas from monadic second-order logic are invariant for bisimulation iff they are definable in the μ-calculus. The dissertation Hollenberg 1997 provides a 'safety companion' to the latter result.

4.3. Invariance and safety with state parameters

The following simple refinement of our results turns out useful in several applications. For instance, in the literature on process logics, labeled transition systems often come with a distinguished 'root' s_0, standing for the starting state of the process:

$$(S, \{R_a\}_{a \in A}), s_0).$$

In the definition of 'bisimulation', one then adds the requirement that the roots be related. The previous results can be extended to this case, too, with the following modifications. First, the modal language is to be enriched to deal with this new feature, by introducing an operator $\mathrm{GOTO}_{\mathrm{root}}\,\varphi$ 'resetting' evaluation to the root:

$$M, s \models \mathrm{GOTO}_{\mathrm{root}}\,\varphi \qquad \text{iff} \qquad M, s_0 \models \varphi.$$

Some obvious valid interchange principles govern the use of this operator (cf. the "Now" operator in temporal logic), allowing some normal forms. An easy adaptation of the proof for the Invariance Theorem to this enriched modal language gives the following result:

THEOREM. *A first-order formula $\varphi(x)$ over labeled transition systems is invariant for root-to-root bisimulation if and only if it is definable by a modal formula using ordinary modalities as well as* $\mathrm{GOTO}_{\mathrm{root}}$.

The new formulas may be used to force two roots to have the same modal types, so that they will stand in the bisimulation constructed. Next, concerning the Safety Theorem, the crucial new addition to our repertoire is the following binary relation of *root resetting*, which is always safe for root-to-root bisimulations:

$$\lambda xy \bullet y = s_0.$$

This is the natural binary relation for the modality $\mathrm{GOTO}_{\mathrm{root}}$. All earlier arguments go through with this addition, which will allow us to enforce a root-to-root bisimulation in the crucial part of the Continuity Lemma of Section 2. Thus, mutatis mutandis, we obtain:

THEOREM. *A first-order relational operation $O(R_1, \ldots, R_n)$ is safe for root-to-root bisimulation if and only if it can be defined using atomic relations $R_a xy$ as well as root resetting, atomic tests $(q)?$ for propositional atoms q in our models, using the three operations* ;, $\cup$ *and* $\sim$.

These outcomes can be generalized to the case where further states become distinguished parameters in labeled transition systems, leading to additional fixed links in our bisimulations. In this case, for each of these states y, one adds a modal operator GOTO_y and a resetting relation RES_y to the syntactic repertoire, just as for the above root.

4.4. Process algebra and respect for bisimulation

Our analysis has been confined to 'internal' description languages for labeled transition systems, with modal statements concerning single states, and dynamic logic programs moving between states. But the literature on Process

Algebra uses 'external' formalisms describing operations creating new LTSs out of old ones (cf. Milner 1980). Examples are operations like *action prefix*, making a process $X = aY$ from Y by adding a new root and joining it to the root of Y by one a-arrow, or *choice* making a process $X = Y + Z$ out of Y, Z by adding a new root and letting it have the union of all initial transitions from the roots of Y, Z. There exists a hierarchy of more complex constructions over LTSs, such as *product*, various forms of *parallel merge* and operators defined by *recursion*. Here too, Safety makes sense. A ubiquitous constraint on process-algebraic operations O is 'respect for bisimulation'. This says that, if Y, Y', Z, Z', ... are bisimilar, then so are $O(Y, Z, \ldots)$ and $O(Y', Z', \ldots)$. But even stronger intuitions are at work here, amounting to this:

> "Any given list of bisimulations for the arguments can be transformed *uniformly* into a bisimulation for the values of the LTS operation."

We can lift the earlier analysis to this area, with a possible hierarchy of notions of safety, depending on what one means by 'uniformity'. The resulting syntactic characterization of Safety would introduce various new formats of definition for process-algebraic operations. (For further discussion of this 'logical space', including a special modal logic for external LTS-operations, cf. van Benthem 1993.) Here, we just state one partial result. Algebraic operations of action prefix and choice may be analyzed inside single LTSs after all, by identifying processes with states (standing for their full 'generated submodels'). Thus, a definition for a new algebraic operation $x = O(y, z, \ldots)$ involves a stipulation, for each atomic action R_a, which successors are going to occur for the new root x. E.g., we have

Action Prefix	$R_axu := u = y$	
	$R_bxu := \perp$	for all b distinct from a
Choice	$R_axu := R_ayu \vee R_azu$	for all atomic actions a

Let us call operations that are first-order definable in this format $R_axu := \delta_a(u, y, z, \ldots)$ *elementary operations*. Moreover, our two sample process operations satisfy

> *Respect for Bisimulation*
> "Each bisimulation on the arguments automatically becomes a bisimulation for the values upon the mere addition of a link between the two new roots."

For a negative example, giving the new root the intersection (rather than the union) of all successors for its argument roots does not respect bisimulation. Now, the (parametrized) Safety Theorem characterizes all algebraic operations in this format.

THEOREM. *The elementary process operations respecting bisimulation are precisely those that can be defined in the format* $R_a xu := \delta_a(y,u)$ *where* $\delta_a(s,u)$ *is a syntactic description of a safe operation, allowing resetting relations for the argument parameters, with 'y' substituted for its first argument.*

PROOF. In one direction, it is easy to see that these forms define strictly safe operations. Conversely, we need a technical observation, saying essentially that the unary predicate satisfied by u given some fixed x has an obvious 'zigzag behaviour' under bisimulation:

CLAIM. A schema of definition δ defines a strictly safe operation if and only if the relation $\lambda su \bullet \delta(u,y,z,\ldots)$ (i.e., 'jump to some successor of the new root') is safe for bisimulation in our earlier sense, using the earlier parametrized format with distinguished states for the relevant argument roots $y, z, \ldots$

Now, we can describe these operations exhaustively via the earlier Safety Theorem, in its parametrized form. Moreover, as the above relation does not depend on its first argument, an arbitrary parameter (say, y) may be substituted for the first variable in the resulting schema of definition $\delta(s,u)$, which yields the promised format of definition. ■

It is easy to see that Action Prefix and Choice indeed fall under the preceding description:

$$\begin{array}{ll} u = y & [y/s]\,\mathrm{RES}_y \\ R_a yu \vee R_a zu & [y/s]\,((\mathrm{RES}_y\,;\,a) \cup (\mathrm{RES}_z\,;\,a)), \end{array}$$

whereas 'Intersection' would need an irreducible conjunction which is beyond this format.

This simple syntactic description is only a first step toward a more ambitious classification of process-algebraic operations. In particular, the latter may involve constructions of new states out of old ones, usually employing ordered *pairs* or *sequences*. Over these, more complex new transition relations may then be defined, e.g. for parallel merges. In response to the first version of this paper, as well as an unpublished follow-up (van Benthem 1993A, 1993B), Marco Hollenberg (1995A, 1997) has produced a number of remarkable extensions and refinements. He generalizes the above notions of safety and respect for bisimulation, and their corresponding modal definability to first-order languages that manipulate tuples of objects up to some fixed length over so-called 'product structures', and then proves very general classification theorems. The resulting format of definition covers all the usual ACP-style process-algebraic operations — while Hollenberg's arguments and outcomes are also of independent modal and model-theoretic interest.

4.5. Invariance, safety and logicality

Safety for bisimulation implies well-known semantic constraints on intuitive 'logicality', such as 'invariance for permutations' of individual domains. Indeed, the style of analysis in this paper suggests a more general view on arbitrary logical operations, as guaranteeing computability within the semantic framework established by some proces equivalence (bisimulation, potential isomorphism, isomorphism, ...). Chapter 5 of van Benthem 1996 has a general discussion of logical constants as process operations, presenting safety for bisimulation as lying at one end of a whole spectrum of semantic invariances to this effect.

Acknowledgment

I would like to thank the participants of various logic colloquia held at Amsterdam and Stanford for their interest in these topics. The same holds for the audience of the ASL Logic Colloquium at Clermont-Ferrand 1994 — and in particular, its excellent organizers Denis Richard and Jacek Tomasik. At a later stage, the audience of my 'dynamics class' at the ESSLLI Summer School in Prague (1996) was very helpful and stimulating. More technically, Larry Moss and Jon Barwise have taught me crucial facts about infinitary extensions of modal and dynamic logic. Finally, I have learnt much from two inventive colleagues at Utrecht who took up the above ideas, namely, Albert Visser — and especially, as will be evident from the current update of Section 4, Marco Hollenberg, whose forthcoming dissertation takes all the above ideas to greater depth and width.

References

J. Barwise & J. van Benthem

1996 'Interpolation, Preservation, and Pebble Games', Report ML-96-12, Institute for Logic, Language and Computation, University of Amsterdam.

J. van Benthem

1976 *Modal Correspondence Theory*, dissertation, Mathematical Institute, University of Amsterdam.

1984 'Correspondence Theory', in D. Gabbay & F. Guenthner, eds., *Handbook of Philosophical Logic*, vol. II, Reidel, Dordrecht.

1986 *Essays in Logical Semantics*, Reidel, Dordrecht, (Studies in Linguistics and Philosophy, vol. 29).

1991 *Language in Action. Categories, Lambdas and Dynamic Logic*, North-Holland, Amsterdam, (Studies in Logic, vol. 130).

1993A 'A Modal Perspective on Process Operations', manuscript, Institute for Logic, Language and Computation, University of Amsterdam.

1993B 'Programming Operations that are Safe for Bisimulation', Report 93-179, Center for the Study of Language and Information, Stanford University.

1995 'Logic and the Flow of Information', in D. Prawitz, B. Skyrms & D. Westerståhl, eds., *Proceedings 9th International Congress of Logic, Methodology and Philosophy of Science. Uppsala 1993*, Elseviers Science Publishers, Amsterdam, 693–724.

1996 *Exploring Logical Dynamics*, Studies in Logic, Language and Information, CSLI Publications (Stanford) & Cambridge University Press.

1997A 'Bits and Pieces', Report LP-97-01, Institute for Logic, Language and Computation, University of Amsterdam.

1997B 'Modality, Bisimulation and Interpolation in Infinitary Logic', to appear in *Annals of Pure and Applied Logic* (K. Georgatos et al., eds., *Festschrift for Rohit Parikh*).

J. van Benthem & J. Bergstra

1995 'Logic of Transition Systems', *Journal of Logic, Language and Information* 3:4, 247–283.

C. C. Chang & H. J. Keisler

1973 *Model Theory*, North-Holland, Amsterdam.

D. Harel

1984 'Dynamic Logic', in D. Gabbay & F. Guenthner, eds., *Handbook of Philosophical Logic*, vol. II, Reidel, Dordrecht.

M. Hollenberg

1995A 'Bisimulation Respecting First-Order Operations', Logic Group Preprint Series 156, Department of Philosophy, University of Utrecht.

1995B 'Bisimulation Safety over Finite Models', manuscript, Department of Philosophy, University of Utrecht.

1996 'Generalized Safety for Bisimulation', in P. Dekker & M. Stokhof, eds., *Proceedings Tenth Amsterdam Colloquium*, Institute for Logic, Language and Computation, University of Amsterdam.

1997 *Modal Logic and Process Algebra*, to appear, Ph.D. dissertation, Philosophical Institute, Rijksuniversiteit Utrecht.

D. Janin & I. Walukiewicz

1996 'On the Expressive Completeness of the Propositional μ-Calculus with Respect to Monadic Second-Order Logic', Department of Mathematics and Informatics, University of Bordeaux & Department of Computer Science, Aarhus University.

H. J. Keisler

1971 *Model Theory for Infinitary Languages*, North-Holland, Amsterdam.

R. Milner

1980 *A Calculus of Communicating Systems*, Springer, Berlin.

D. Park

1981 'Concurrency and Automata on Infinite Sequences', *Proceedings 5th GI Conference*, Springer, Berlin, 167–183.

E. Rosen

1995 'Modal Logic over Finite Stuctures', Report ML-95-08, Institute for Logic, Language and Computation, University of Amsterdam. To appear in the *Journal of Logic, Language and Information.*

Johan van Benthem
Institute for Logic, Language and Computation
University of Amsterdam