

7-1996

The Behavior-Realization Adjunction and Generalized Homomorphic Relations

Gary T. Leavens
Iowa State University

Don Pigozzi
Iowa State University

Follow this and additional works at: http://lib.dr.iastate.edu/cs_techreports



Part of the [Systems Architecture Commons](#), and the [Theory and Algorithms Commons](#)

Recommended Citation

Leavens, Gary T. and Pigozzi, Don, "The Behavior-Realization Adjunction and Generalized Homomorphic Relations" (1996).
Computer Science Technical Reports. Paper 128.
http://lib.dr.iastate.edu/cs_techreports/128

This Article is brought to you for free and open access by the Computer Science at Digital Repository @ Iowa State University. It has been accepted for inclusion in Computer Science Technical Reports by an authorized administrator of Digital Repository @ Iowa State University. For more information, please contact digirep@iastate.edu.

The Behavior-Realization Adjunction and Generalized Homomorphic Relations

Gary T. Leavens and Don Pigozzi
TR #94-18b

September 1994, revised September 1994, July 1996

Keywords: behavior, realization, observable equivalence, simulation, generalized relation, abstract data type, model theory.

1994 CR Categories: D.3.3 [*Programming Languages*] Language Constructs — Abstract data types; F.3.2 [*Logics and Meanings of Programs*] Semantics of Programming Languages — algebraic approaches to semantics; F.3.2 [*Mathematical Logic and Formal Languages*] Mathematical Logic — model theory.

1991 Mathematics Subject Classification. Primary: 68Q65 Secondary: 68N05, 68N15, 68Q60.

A slightly abbreviated version will appear in *Theoretical Computer Science*.

Copyright © Gary T. Leavens and Don Pigozzi, 1994, 1996. This copyright will be transferred to the journal.

Department of Computer Science
226 Atanasoff Hall
Iowa State University
Ames, Iowa 50011-1040, USA

THE BEHAVIOR-REALIZATION ADJUNCTION AND GENERALIZED HOMOMORPHIC RELATIONS

GARY T. LEAVENS AND DON PIGOZZI

Iowa State University

July 30, 1996

ABSTRACT. A model theory for proving correctness of abstract data types is developed within the framework of the behavior-realization adjunction. To allow for incomplete specifications, proof-of-correctness is based on comparison to one of several paradigmatic models. For making such comparisons, the notions of the behavior and realization relations, and their duals are developed. These relations are used to give the first exact algebraic characterization of behavioral reduction and equivalence for algebras that are not term-generated.

INTRODUCTION

The main advantage of abstract data types (ADTs) in programming is that they allow reasoning at an appropriate level. In reasoning about code that uses an ADT, clients rely on the ADT's specification, instead of using more complex and overly specific reasoning about the ADT's implementation. The soundness of such an abstract reasoning technique means that if an implementation is certified correct, then its visible behavior will not be surprising. By *visible behavior* we mean, informally, the printed or returned results of programs. By *surprising behavior* we mean visible behavior that would contradict the predictions of the specification. Completeness of an abstract reasoning technique means that if an implementation cannot exhibit surprising behavior, then it can be certified as correct.

We investigate sound and complete model-theoretic techniques for proving that a candidate implementation of an ADT is correct. For reasons discussed below, we are especially interested in specifications that are incomplete and not term-generated. For us, a complete specification is one for which all of its models are behaviorally equivalent, and a specification is term-generated if there are nonvisible types that fail to have a complete system of constructors. We shall also assume that a candidate implementation has already been adapted to the interface (signature) required ("derived" in the sense of Section 5.5 of [8]).

What is known about the soundness and completeness of techniques for proving that a candidate implementation of an ADT is correct? We shall restrict ourselves here to model-theoretic methods. Previous model-theoretic work on this problem,

The work of both authors is based upon work supported by the National Science Foundation under Grants CCR-9593168 and also CCR-9108654 (for Leavens) and DMS-8805870 (for Pigozzi).

like our work, is based on comparisons to paradigmatic models. In most work, there is only one paradigmatic model mentioned, and so the ADT’s specification must be complete. If the specification is incomplete, there is no way to choose a single paradigm, and the technique must be adapted somehow to deal with the choice of an appropriate paradigm before the comparison. However, it is a simple matter to adapt this technique to incomplete specifications by using a collection of paradigms. These paradigms collectively span the permitted behaviors, and thus to prove the correctness of a candidate implementation, one must first choose a paradigm and then make the comparison.

This paper concentrates, therefore, on how to compare an implementation algebra to a paradigm, once a paradigm is selected. Several authors have studied such notions previously (including [4, 7, 17, 18] — see Section 8 of [23] for a survey). For our purposes the most important technique is that of Schoett [18]. Schoett casts the problem as one of showing that a partial algebra \mathbf{A} can be used in place of the paradigm, a partial algebra \mathbf{B} , without exhibiting surprising behavior. He argues that this will be assured if the two algebras are behaviorally equivalent in the sense that any program that is run in the two algebras has the same output. He makes the natural assumption that only visible data is legitimate input-output for the program. He proves that the existence of a bisimulation between \mathbf{A} and \mathbf{B} , i.e., a homomorphic relation that is the identity on visible types, is both necessary and sufficient for the behavior of \mathbf{A} to be equivalent to the behavior of \mathbf{B} .

It can be argued however that Schoett’s criterion for behavioral equivalence is not restrictive enough. It fails to detect some behavioral differences that an ADT implementor might care about. The main problem with his approach is that programs can take only visible data as input and hence algebras can be compared only with respect to the behavior of visible data. For example, in the context of specifying a parameterized type (e.g., a parameterized priority-queue), consider the specification of its formal type parameter, $\mathbf{P0}$. The only operation that would be specified for $\mathbf{P0}$ would be a comparison predicate, $\mathbf{1eq}$, taking two $\mathbf{P0}$ s and returning a Boolean; no constructors would be specified for $\mathbf{P0}$. In this example, the type $\mathbf{P0}$ would not be a visible type (i.e., it could not be directly input or output). Hence the only visible type in the example is the Booleans, and the type $\mathbf{P0}$ is hidden. Because $\mathbf{P0}$ is hidden and there are no constructors for it, programs with visible input-output cannot make any interesting observations. Hence, using Schoett’s criterion, even candidate implementations that, say, fail antisymmetry would be certified as correct. In this paper we adapt Schoett’s technique by considering not just observations with visible inputs, but “procedures” with nonvisible inputs. For example, this allows us to make behavioral distinctions in the $\mathbf{P0}$ example. That is, we allow the behavior of nonvisible data to be compared in different models, leading to a stronger notion of implementation which is important in situations where the specification is not term-generated.

ADTs that are not term-generated are even more important for object-oriented programming than they are in more conventional programming with ADTs. For example, a library of object-oriented ADTs typically includes a type **Collection** that is “abstract” in the sense that it has no constructors. Such a type will have subtypes such as **Set**, **Bag**, **List**, and **Array**. Existing objects of one of these subtypes can be treated as if they were collections. This is analogous to the way

that objects having the type of a formal type parameter, such as $\mathbf{P0}$, are treated in parameterized code [21]. It is also apparent from this example why it is important to be able to compare nonvisible data. It is natural to want to compare the behavior of a bag constructed from the integers 1, 2, and 3, for instance, with that of a set constructed from the same integers. But this cannot always be achieved by simply comparing the behavior of the visible data, such as the integers 1, 2, and 3 in two different models, because a (deterministic) program with only visible input would construct either a set in both \mathbf{A} and \mathbf{B} or a bag in both \mathbf{A} and \mathbf{B} , but not a set in \mathbf{A} and a bag in \mathbf{B} . This problem is the original motivation for our study of “procedures” with nonvisible inputs.

In this paper, we give a sound and complete algebraic technique for proving the correctness of an implementation, which need not be term-generated. The technique uses a general notion of simulation, which in turn uses a generalization of the notion of homomorphic relation; such a generalization is necessary because standard homomorphic relations do not give a complete characterization technique for specifications that are not term-generated.

The idea that motivates the definition of a generalized homomorphic relation is simple. We want to capture exactly those distinctions procedures can make so that, if no surprises arise from the use of data elements, because of the incompleteness of a specification, then there is a generalized simulation. Put another way, if the differences that may exist between the string $\vec{a} = a_0 \cdots a_{n-1}$ of elements of \mathbf{A} and a corresponding string $\vec{b} = b_0 \cdots b_{n-1}$ of elements of the paradigm \mathbf{B} cannot be detected by any program, then a generalized simulation of \mathbf{B} by \mathbf{A} exists that correlates, not only \vec{a} with \vec{b} , but also any pair of strings with the same property. It might be surprising to learn that there may exist no standard simulation of this kind in this situation; an example of this phenomenon can be found in the Appendix. A standard simulation, by definition, correlates single pairs of elements and consequently can correlate two strings $a_0 \cdots a_{n-1}$ and $b_0 \cdots b_{n-1}$ only by correlating a_i with b_i , for each $i < n$. This means that correlations determined by a standard simulation are additive in the following sense: if $a_1 \cdots a_n$ and $a'_1 \cdots a'_n$ are correlated respectively with $b_1 \cdots b_n$ and $b'_1 \cdots b'_n$, then by necessity $a_1 \cdots a_n a'_1 \cdots a'_n$ and $b_1 \cdots b_n b'_1 \cdots b'_n$ must be correlated. The problem is that, while there may be no observation that behaves differently when given $a_1 \cdots a_n$ as inputs as when given $b_1 \cdots b_n$ as inputs, and similarly for $a'_1 \cdots a'_n$ and $b'_1 \cdots b'_n$, it is quite possible that there is a program that behaves differently with $a_1 \cdots a_n a'_1 \cdots a'_n$ and $b_1 \cdots b_n b'_1 \cdots b'_n$ as inputs. Therefore, generalized relations correlate whole strings of inputs rather than the individual members of the strings. Actually, it is more convenient from a technical point of view to think of generalized relations as correlating environments, that is, assignments of variables to elements, rather than strings. Consequently, while standard homomorphic relations are indexed by types, a generalized homomorphic relation is a family of relations indexed by type contexts. This is analogous to the standard kind of relation indexed by types if one thinks of a type context as the “type” of an environment; more formally, a *type context* gives the type of each variable in an environment, and is thought of as a map from variables to types.

One way to see the power of generalized relations is by using an analogy between an environment and an algebra extended with new constants. The new constants

are analogous to the variables in the domain of the environment. Comparing environments is thus akin to comparing such extended algebras, and the extended algebras allow what were formerly unnamed elements of the algebra's carrier set to be named. Taken to the extreme, such an extension of an algebra that is not term-generated would be term-generated, but would, in general, require infinitely many new constants. Instead of using infinitely many new constants added to an algebra, one can consider infinitely many such finite extensions; that is, one can consider all environments over such algebras. Because standard relations give a complete characterization technique for term-generated algebras, one can see by this analogy how generalized relations should give (and as we will show, do give) a complete characterization technique for algebras that are not term-generated.

We have found it useful to adapt the concepts of “behavior” and “realization” as they are developed in Goguen and Meseguer [5, 6] for the formalism in which to present our results. Formally we think of standard relations as the “behavior” of generalized relations and generalized relations as the “realization” of standard ones. To explain, suppose \mathbf{A} and \mathbf{B} are algebras, a is an element of \mathbf{A} and b element of \mathbf{B} of the same type. Suppose, in addition, that ρ is an environment over \mathbf{A} and σ is a similar environment over \mathbf{B} . The pairs $\langle a, b \rangle$ and $\langle \rho, \sigma \rangle$ are “behavior-and-realization” related if there is a procedure P in the observational language such that P , when run in the environment ρ , has output a , and when P is run in the environment σ it has output b . The pair $\langle a, b \rangle$ is thought of as a part of the “behavior” of $\langle \rho, \sigma \rangle$, and $\langle \rho, \sigma \rangle$ in turn is thought of as a partial “realization” of the behavior $\langle a, b \rangle$. (Returning to the analogy between environments and extended algebras, a realization can be thought of as an extension of the algebra which gives the specified output for P .) This gives two maps, one from the lattice of generalized relations (under set-theoretic inclusion) to the lattice of standard relations, and one in the opposite direction, that form a Galois connection, i.e., an adjunctive pair of functors between the two lattices viewed as simple categories. This adjunction is the basis of our model-theoretic approach to implementation correctness.

By a *behavior* we mean a standard relation between two models \mathbf{A} and \mathbf{B} of the specification that is the behavior of some generalized relation and a *realization* is the realization of some standard relation.¹ At the center of the theory are the following questions: under what circumstances is a behavior a standard homomorphic relation (and vice-versa), and under what circumstances is a realization a generalized homomorphic relation (and vice-versa)? The main results presented in this paper are the following: every standard homomorphic relation is a behavior (Thm. 4.3) and without qualification, every realization is a generalized homomorphic relation (Thm. 4.10).

We also give a new proof of Schoett's theorem that firmly places it within our general framework (Thm. 4.9). Finally, we extend Schoett's technique to deal with a more refined notion of behavioral equivalence in which the behavior of nonvisible data is considered (Thm. 4.13).

The rest of the paper is organized as follows. Sec. 1 quickly reviews basic ter-

¹The term *realization* is intended to suggest the realization of a specification in the sense of a correct implementation of the specification [5, 6]. We think of a standard relation as specifying the behavior of a pair of environments and the environment pairs realizing the behavior as the correct implementation of the specification.

minology. Sec. 2 defines relations between environments (generalized relations). In Sec. 3 the behavior-realization adjunction is developed. Homomorphic relations, both standard and generalized are discussed in Sec. 4; the principal results of the paper are included in this section. Sec. 5 contains some conclusions and a discussion of future and related work. Finally, in the Appendix we show by example that standard simulations, of the type used by Schoett, cannot be used to characterize our stronger notion of behavioral equivalence. In the latter part of the Appendix we explore in some detail the exact connection between standard homomorphic relations and generalized homomorphic relations.

1 PRELIMINARIES

In this section we review the notation and terms needed in the rest of the paper.

Signatures are hierarchical over a set of visible types and allow for the overloading of operations [16]. Let $\mathbb{N} = \{0, 1, 2, 3, \dots\}$ be the set of natural numbers.

Definition 1.1 (signature). A signature $\Sigma = \langle TYPE, VIS, VAR, OP, ResType \rangle$ consists of:

- (i) A set $TYPE$ of *types* together with a subset VIS of *visible types*.
- (ii) A countably infinite set VAR of variable symbols.
- (iii) A \mathbb{N} -indexed family $OP = \langle OP_n : n \in \mathbb{N} \rangle$ of *operation symbols*, where OP_0 is nonempty.
- (iv) A \mathbb{N} -indexed family $ResType = \langle ResType_n : n \in \mathbb{N} \rangle$ of partial functions with $ResType_n : OP_n \times TYPE^n \rightarrow TYPE$ for each $n \in \mathbb{N}$. \square

Example 1.2. A signature, Σ_{P0} , for algebras with the partial order type, $P0$, is defined as follows:

$$\begin{array}{ll}
 TYPE_{P0} := \{\text{Bool}, P0\}, & ResType_0(\text{true}, \langle \rangle) = \text{Bool}, \\
 VAR_{P0} := \{x_1, x_2, \dots\}, & ResType_0(\text{false}, \langle \rangle) = \text{Bool}, \\
 VIS_{P0} := \{\text{Bool}\}, & ResType_1(\text{not}, \langle \text{Bool} \rangle) = \text{Bool}, \\
 OP_0 := \{\text{true}, \text{false}\}, & ResType_2(\text{and}, \langle \text{Bool}, \text{Bool} \rangle) = \text{Bool}, \\
 OP_1 := \{\text{not}\}, & ResType_2(\text{or}, \langle \text{Bool}, \text{Bool} \rangle) = \text{Bool}, \\
 OP_2 := \{\text{and}, \text{or}, \text{leq}\}, & ResType_2(\text{leq}, \langle P0, P0 \rangle) = \text{Bool}. \\
 OP_n := \{\}, \text{ for } n > 2, &
 \end{array}$$

Σ_{P0} is the signature of a class of algebras for which $P0$ can be viewed as a formal parameter type of the kind mentioned in the Introduction. \square

Suppose n is the rank of g , $\vec{T} \in TYPE^n$, and $ResType(g, \vec{T}) = S$. The pair $\langle \vec{T}, S \rangle$ is called a *type* of g and will be written $\vec{T} \rightarrow S$; \vec{T} is called a *type domain* of g and S the *result type* of g for \vec{T} . Due to operation overloading, an operation symbol may have many types but at most one result type for each type domain. Let ε denote the empty string. If g is a constant, we identify $\varepsilon \rightarrow S$ with S and type with result type. The operation g is *trivial* if its set of types is empty, i.e., if

$ResType(g, \vec{T})$ is undefined for all $\vec{T} \in TYPE^n$. We assume Σ contains no trivial operations.

With each signature we associate a unique subsignature of visible types for the purpose of defining observations over Σ .

Definition 1.3 (visible subsignature). Assume $\Sigma = \langle TYPE, VIS, VAR, OP, ResType \rangle$ is a signature. The *visible subsignature* of Σ ,

$$\Sigma_{VIS} = \langle TYPE|_{VIS}, VIS, VAR, OP|_{VIS}, ResType|_{VIS} \rangle,$$

is defined as follows. $TYPE|_{VIS} = VIS$ and $OP|_{VIS}$ is the set of all operations in OP whose restriction to VIS is nontrivial. For $g \in OP_n$ and $\vec{V} \in VIS^n$,

$$ResType|_{VIS}(g, \vec{V}) = \begin{cases} ResType(g, \vec{V}), & \text{if } ResType(g, \vec{V}) \in VIS \\ \text{undefined,} & \text{otherwise. } \square \end{cases}$$

For the signature of Ex. 1.2, $TYPE_{P0}|_{VIS} = \{\text{Bool}\}$, $OP_2|_{VIS} = \{\text{and, or}\}$, and $ResType|_{VIS}$ differs from $ResType$ only in being undefined when leq is the first argument.

Σ -terms are formed from a signature in the usual way. That is, every variable and constant is a term, and, if $g \in OP_n$ (with $n \geq 1$) and t_1, \dots, t_n are terms, then $g(t_1, \dots, t_n)$ is a term. A term is *ground* if it contains no variables. By assumption the set of ground Σ -terms is nonempty. We write $t(x_1, \dots, x_n)$ for a term t when we want to indicate that the variables actually occurring in t must be in the list x_1, \dots, x_n . In this context $t(s_1, \dots, s_n)$ denotes the result of simultaneously substituting the terms s_1, \dots, s_n respectively for x_1, \dots, x_n .

Definition 1.4 (type context). A finite set H of the form $\{ \langle x_1, T_1 \rangle, \dots, \langle x_n, T_n \rangle \}$, where x_1, \dots, x_n are distinct variables and $T_1, \dots, T_n \in TYPE$ is called a *type context*; i.e., a type context is a finite function from variables to types. The set of variables $\{x_1, \dots, x_n\}$ of H is denoted by $Dom(H)$ and T_i is denoted by $H(x_i)$. H is *visible* if the type of every variable in H is visible. The set of all type contexts is denoted by $TCON$ and the set of all visible type contexts by $TCON|_{VIS}$. K is a *subcontext* of H if $Dom(K) \subseteq Dom(H)$ and $K(x) = H(x)$ for all $x \in Dom(K)$. \square

The type inference rules for this grammar are given below.

$$\begin{aligned} \text{(ident)} \quad & \Sigma; H \vdash x : T, \quad \text{if } H(x) = T, \\ \text{(op-call)} \quad & \frac{\Sigma; H \vdash \vec{t} : \vec{T}}{\Sigma; H \vdash g(\vec{t}) : S}, \quad \text{if } ResType(g, \vec{T}) = S \in TYPE. \end{aligned}$$

When we write $\Sigma; H \vdash t : T$ we mean that this sequent can be proved by applying the above rules. In this case T is unique and is called the *H-type* of t .

We say that t is *well H-typed* if it has a H -type. When Σ is clear from context we write $H \vdash t : T$. When the type context H is also clear we may speak of “the type” of t and of t being “well-typed”. We often identify the type-expression “ $x : T$ ” with the ordered pair $\langle x, T \rangle$. Thus we will denote the extended type context $H \cup \{ \langle x, T \rangle \}$ by $H, x : T$. We further streamline notation by using the expression “ $t : T$ ” when referring to a term t , with the understanding that this automatically entails the assumption t is well-typed and of type T . We write $\vdash \vec{t} : \vec{T}$ as shorthand for $\Sigma \vdash t_1 : T_1, \dots, \Sigma \vdash t_n : T_n$. Similar vector abbreviations will be used below without further explanation.

Definition 1.5 (context homomorphism, homomorphic pre-image). Let H and K be type contexts. A mapping $h : \text{Dom}(K) \rightarrow \text{Dom}(H)$ is said to be a *context homomorphism* from K to H if $K \vdash x : T$ implies $H \vdash h(x) : T$ for every $x \in \text{VAR}$. K is called a *homomorphic pre-image* (or simply a *pre-image*) of H under h . \square

If h is a context homomorphism from K to H and $x_1, \dots, x_n \in \text{Dom}(K)$, then for every term $t(x_1, \dots, x_n)$ and type T

$$K \vdash t(x_1, \dots, x_n) : T \quad \text{iff} \quad H \vdash t(h(x_1), \dots, h(x_n)) : T.$$

Σ -algebras. Models of abstract data types with signature Σ are called Σ -algebras. These models have interpretations for operations that are polymorphic in that they directly model overloaded operations.

Definition 1.6 (Σ -algebra). A Σ -algebra $\mathbf{A} = \langle A, \{g^{\mathbf{A}} : g \in OP\} \rangle$ consists of:

- (i) A *TYPE*-indexed family of sets, $A = \langle A_T : T \in \text{TYPE} \rangle$, called the *carrier of \mathbf{A}* .
- (ii) A partial function, $g^{\mathbf{A}} : (\bigcup_{S \in \text{TYPE}} A_S)^n \rightarrow \bigcup_{S \in \text{TYPE}} A_S$, for each $n \in \mathbb{N}$ and $g \in OP_n$, called the *interpretation of g* , with the property that, for every type $T_1 \cdots T_n \rightarrow S$ of g and every $a_1 \cdots a_n \in A_{T_1} \times \cdots \times A_{T_n}$, $g^{\mathbf{A}}(a_1, \dots, a_n)$ is defined and contained in A_S . \square

Example 1.7. Let Σ_{P0} be the signature of Ex. 1.2. The Σ_{P0} -algebra **INT** is defined as follows.

$$\begin{aligned} \text{INT}_{\text{Boo1}} &:= \{tt, ff\}, \\ \text{INT}_{\text{P0}} &:= \mathbb{Z} = \{\dots, -3, -2, -1, 0, 1, 2, 3, \dots\}, \end{aligned}$$

$$\begin{aligned} \text{true}^{\text{INT}}(\langle \rangle) &= tt, \\ \text{false}^{\text{INT}}(\langle \rangle) &= ff, \\ \text{not}^{\text{INT}}(\langle b \rangle) &= \neg b, \\ \text{and}^{\text{INT}}(\langle b_1, b_2 \rangle) &= b_1 \wedge b_2, \\ \text{or}^{\text{INT}}(\langle b_1, b_2 \rangle) &= b_1 \vee b_2, \\ \text{leq}^{\text{INT}}(\langle n_1, n_2 \rangle) &= n_1 \leq n_2. \quad \square \end{aligned}$$

Example 1.8. The Σ_{P0} -algebra **NAT** is the same as **INT**, except that it has a different carrier for the type **P0**:

$$\text{NAT}_{\text{P0}} := \mathbb{N} = \{0, 1, 2, 3, \dots\}. \quad \square$$

Definition 1.9 (VIS-reduct). The *VIS-reduct* of a Σ -algebra \mathbf{A} is the Σ_{VIS} -algebra

$$\mathbf{A}|_{VIS} = \langle A|_{VIS}, \{g^{\mathbf{A}|_{VIS}} : g \in OP_{VIS}\} \rangle,$$

where $A|_{VIS} = \langle A_V : V \in VIS \rangle$ and $g^{\mathbf{A}|_{VIS}}(\vec{a}) = g^{\mathbf{A}}(\vec{a})$ for every type domain \vec{V} of g consisting only of visible types and every $\vec{a} \in A_{\vec{V}}$. \square

Definition 1.10 (H-environment). Let \mathbf{A} be a Σ -algebra and H a type context. An *H-environment* ρ over \mathbf{A} is a mapping of the variables of the domain of H into $\bigcup_{S \in TYPE} A_S$ such that $\rho(x) \in A_T$ iff $H \vdash x : T$. The set of all *H-environments* in \mathbf{A} is denoted by $ENV_{\mathbf{A}}^H$. An *H-environment* is *visible* if H is visible. \square

When H is clear from context we simply call these “environments”.

Let ρ be an *H-environment* and assume $y \in VAR \setminus Dom(H)$. Let $T \in TYPE$ and $a \in A_T$. The $(H, y : T)$ -environment that assigns a to y and $\rho(x)$ to each variable x of H will be denoted by $[y \mapsto a]\rho$.

The notions of a subenvironment and pre-image of an environment under a context homomorphism are defined in the obvious way. In particular, if ρ is a *H-environment* and $h : K \rightarrow H$ is a context homomorphism, then pre-image of ρ is the *K-environment* $\rho \circ h$.

Definition 1.11 (meaning). Let H be a type context, $t : T$ a well *H*-typed term, and ρ an *H-environment*. Then the *meaning* $\llbracket t : T \rrbracket^{\mathbf{A}} \rho$ of t under ρ is defined by recursion on the structure of t in the usual way:

$$\begin{aligned} \llbracket x : T \rrbracket^{\mathbf{A}} \rho &= \rho(x), \quad \text{for each variable } x : T. \\ \llbracket g(s_1, \dots, s_n) : T \rrbracket^{\mathbf{A}} \rho &= g^{\mathbf{A}}(\llbracket s_1 : S_1 \rrbracket^{\mathbf{A}} \rho, \dots, \llbracket s_n : S_n \rrbracket^{\mathbf{A}} \rho), \\ &\quad \text{for all } g \in OP_n \text{ and all terms } s_1, \dots, s_n. \quad \square \end{aligned}$$

If $H \vdash t(x_1, \dots, x_n) : T$, $\rho \in ENV_{\mathbf{A}}^H$, and $\rho(x_1) = a_1, \dots, \rho(x_n) = a_n$, then we write $t^{\mathbf{A}}(a_1, \dots, a_n)$ in place of $\llbracket t : T \rrbracket^{\mathbf{A}} \rho$.

A Σ -algebra is *term-generated* if every element is denoted by $t^{\mathbf{A}}$ for some ground term t .

We are not concerned in this paper with formal specifications, but only with comparing a candidate implementation (an algebra) with a paradigm (another algebra). The following definition specifies those algebras that can be sensibly compared in the context of this paper.

Definition 1.12 (comparable algebras). Algebras \mathbf{A} and \mathbf{B} are *comparable* if

- (i) \mathbf{A} and \mathbf{B} are both Σ -algebras, for some Σ .
- (ii) $\mathbf{A}|_{VIS} = \mathbf{B}|_{VIS}$.
- (iii) $\mathbf{A}|_{VIS}$ is term-generated. \square

Requiring the visible reducts of both algebras to be equal results in a slight loss of generality—it is enough to have the visible reducts be isomorphic. But the requirement simplifies the following discussion, as the isomorphism can be ignored.

In what follows all pairs of algebras mentioned in the same context are assumed to be comparable unless explicitly indicated otherwise.

2 RELATIONS BETWEEN ALGEBRAS

Simulation between comparable algebras is formalized as a binary relation between the carriers of the two algebras with certain properties. We consider two kinds of relations between algebras, and thus two kinds of simulation. A standard relation relates individual elements of the algebras. The new results in this paper use a generalized relation that relates environments.

Definition 2.1 (standard and generalized relations). Let \mathbf{A} and \mathbf{B} be (not necessarily comparable) Σ -algebras.

- (i) By a *standard relation* between \mathbf{A} and \mathbf{B} we mean a *TYPE*-indexed family of binary relations $\mathcal{R} := \langle \mathcal{R}_T : T \in \text{TYPE} \rangle$ such that $\mathcal{R}_T \subseteq A_T \times B_T$ for every $T \in \text{TYPE}$.
- (ii) By a *generalized relation* between \mathbf{A} and \mathbf{B} we mean a *TCON*-indexed family of binary relations $\mathcal{G} := \langle \mathcal{G}_H : H \in \text{TCON} \rangle$ such that $\mathcal{G}_H \subseteq \text{ENV}_H^{\mathbf{A}} \times \text{ENV}_H^{\mathbf{B}}$ for every $H \in \text{TCON}$. \square

We sometimes shorten “standard relation” to just “relation”.

Recall that the Cartesian product of $\mathbf{A} \times \mathbf{B}$ is the Σ -algebra whose carrier is the *TYPE*-indexed set $A \times B = \langle A_T \times B_T : T \in \text{TYPE} \rangle$. Thus $(A \times B)_T = A_T \times B_T$ by definition and the standard relations between \mathbf{A} and \mathbf{B} can be identified with the indexed subsets of $A \times B$. Although $\text{ENV}_H^{\mathbf{A}} \times \text{ENV}_H^{\mathbf{B}}$ is not equal to $\text{ENV}_H^{\mathbf{A} \times \mathbf{B}}$, there is a natural correspondence between these two sets, given by $\langle \rho, \sigma \rangle \mapsto \rho \times \sigma$ where $(\rho \times \sigma)(x) = \langle \rho(x), \sigma(x) \rangle$ for every $x \in \text{Dom}(H)$. In the sequel we will identify the H -indexed sets $\text{ENV}^{\mathbf{A}} \times \text{ENV}^{\mathbf{B}}$ and $\text{ENV}_H^{\mathbf{A} \times \mathbf{B}}$. Thus the generalized relations between \mathbf{A} and \mathbf{B} are identified with the indexed subsets of $\text{ENV}^{\mathbf{A} \times \mathbf{B}}$.

Let \mathcal{R} and \mathcal{S} be standard relations between \mathbf{A} and \mathbf{B} and between \mathbf{B} and \mathbf{C} , respectively. The *composition*, $\mathcal{R}; \mathcal{S}$, is the standard relation between \mathbf{A} and \mathbf{C} such that, for each $T \in \text{TYPE}$,

$$a (\mathcal{R}; \mathcal{S})_T c \text{ iff there is some } b \in B_T \text{ such that } a \mathcal{R}_T b \text{ and } b \mathcal{S}_T c, \text{ for all } a \in A_T, c \in C_T.$$

Let \mathcal{R} be a standard relation between \mathbf{A} and \mathbf{B} . The *converse* $\check{\mathcal{R}}$ is the relation between \mathbf{B} and \mathbf{A} such that, for each $T \in \text{TYPE}$,

$$b \check{\mathcal{R}} a \text{ iff } a \mathcal{R} b, \text{ for all } a \in A_T, b \in B_T.$$

The composition and converse of generalized relations are defined similarly

By a standard *visible* relation between \mathbf{A} and \mathbf{B} we mean a standard relation between $\mathbf{A}|_{\text{VIS}}$ and $\mathbf{B}|_{\text{VIS}}$. Given any standard relation \mathcal{R} between \mathbf{A} and \mathbf{B} , by the *visible part* of \mathcal{R} we mean the visible relation $\mathcal{R}|_{\text{VIS}} = \langle \mathcal{R}_V : V \in \text{VIS} \rangle$. The corresponding notions for generalized relations are defined in the obvious way. The special standard and generalized relations given in the following definitions will be especially useful in the sequel.

Definition 2.2 (visible identity relations). Let \mathbf{A} and \mathbf{B} be comparable Σ -algebras.

- (i) The *visible standard identity relation*, $\mathcal{I} \subseteq (A \times B)_{VIS}$, is defined by $\mathcal{I} := \langle \mathcal{I}_V : V \in TYPE|_{VIS} \rangle$, where $\mathcal{I}_V = \{ \langle a, a \rangle : a \in A_V \}$ for all $V \in VIS$.
- (ii) The *visible generalized identity relation*, $\mathcal{I}^+ \subseteq ENV^{(\mathbf{A} \times \mathbf{B})|_{VIS}}$, is defined by $\mathcal{I}^+ = \langle \mathcal{I}_H^+ : H \in TCON|_{VIS} \rangle$, where $\mathcal{I}_H^+ = \{ \langle \rho, \rho \rangle : \rho \in ENV_H^{\mathbf{A}}|_{VIS} \}$. \square

Note that \mathcal{I} and \mathcal{I}^+ can also be considered as relations between \mathbf{A} and \mathbf{B} under the assumption that $\mathcal{I}_T = \emptyset$ when $T \notin VIS$ and $\mathcal{I}_H = \emptyset$ when $H \notin TCON|_{VIS}$.

The passage from \mathcal{I} to \mathcal{I}^+ is a special case of the following general method of constructing a generalized relation from a standard one.

Definition 2.3 (pointwise extension). Let \mathcal{R} be a standard between \mathbf{A} and \mathbf{B} . For each type context $H = \{x_1 : T_1, \dots, x_n : T_n\}$, define $\mathcal{R}_H^+ \subseteq ENV_H^{\mathbf{A} \times \mathbf{B}}$ by the condition

$$\rho \mathcal{R}_H^+ \sigma \quad \text{iff} \quad \rho(x_i) \mathcal{R}_{T_i} \sigma(x_i) \text{ for all } i = 1, \dots, n,$$

and set $\mathcal{R}^+ := \langle \mathcal{R}_H^+ : H \in TCON \rangle$. \mathcal{R}^+ is called the *pointwise extension* of \mathcal{R} . \square

Conversely, every generalized relation \mathcal{G} restricts to a standard relation \mathcal{G}^- in the following way.

Definition 2.4 (projective restriction). Let \mathcal{G} be a generalized relation between \mathbf{A} and \mathbf{B} . For each type T , define $\mathcal{G}_T^- \subseteq A \times B$ by the condition

$$a \mathcal{G}_T^- b \quad \text{iff} \quad \text{there exist } H \in TCON, \rho, \sigma \in ENV_H^{\mathbf{A} \times \mathbf{B}}, \text{ and } x \text{ with } H \vdash x : T \text{ such that } \rho(x) = a, \sigma(x) = b, \text{ and } \rho \mathcal{G}_H \sigma.$$

Set $\mathcal{G}^- := \langle \mathcal{G}_T^- : T \in TYPE \rangle$. \mathcal{G}^- is called the *projective restriction* of \mathcal{G} . \square

The following useful result is easily verified on the basis of the above definitions.

Lemma 2.5. $\mathcal{R}^{+-} = \mathcal{R}$ for every standard relation \mathcal{R} and $\mathcal{G}^{-+} \supseteq \mathcal{G}$ for every generalized relation. \square

In the next section we show how the behavior of an environment gives rise to a different way of associating standard and generalized relations that will prove to be even more useful.

Homomorphic relations.

Standard or generalized relations between Σ -algebras that are preserved under the operations of an algebra, in a sense made precise in the following definitions, are called *homomorphic relations*.² The various notions of one data structure *simulating* another are defined in terms of relations of this kind.

²Homomorphic relations are called *logical relations* when extended to higher types [20]. An independent generalization of logical relations that appears to be closely related to our notion of homomorphic generalized relations is considered in [10].

Definition 2.6 (standard homomorphic relation). Let \mathbf{A} and \mathbf{B} be (not necessarily comparable) Σ -algebras and let \mathcal{R} be a standard relation between \mathbf{A} and \mathbf{B} . \mathcal{R} is a *standard homomorphic relation*, or simply *homomorphic*, if it satisfies the following condition:

(SHR1) For every $g \in OP_n$ and type $\vec{T} \rightarrow S$ of g ,

$$a_1 \mathcal{R}_{T_1} b_1, \dots, a_n \mathcal{R}_{T_n} b_n \text{ implies } g^{\mathbf{A}}(a_1, \dots, a_n) \mathcal{R}_S g^{\mathbf{B}}(b_1, \dots, b_n). \quad \square$$

Definition 2.7 (generalized homomorphic relation). Let \mathbf{A} and \mathbf{B} be Σ -algebras and let \mathcal{G} be a generalized relation between \mathbf{A} and \mathbf{B} . \mathcal{G} is a *generalized homomorphic relation*, or simply *homomorphic*, if the following conditions hold:

(GHR1) Let H be a type context and let $\vec{x} \in Dom(H)^n$ and $\vec{T} \in TYPE^n$ be such that $H \vdash \vec{x} : \vec{T}$. Let $g \in OP_n$ and let type $\vec{T} \rightarrow S$ be a type of g . Then for any $y \in VAR \setminus Dom(H)$ and any pair of H -environments ρ and σ in \mathbf{A} and \mathbf{B} , respectively,

$$(2.1) \quad \rho \mathcal{G}_H \sigma \text{ implies } [y \mapsto g^{\mathbf{A}}(\rho(\vec{x}))]\rho \mathcal{G}_{H,y,S} [y \mapsto g^{\mathbf{B}}(\sigma(\vec{x}))]\sigma.$$

(GHR2) For all type contexts H and K and every context homomorphism h from K to H ,

$$\rho \mathcal{G}_H \sigma \text{ implies } \rho \circ h \mathcal{G}_K \sigma \circ h. \quad \square$$

The following property of generalized homomorphic relations is an immediate consequence of (GHR2).

(GHR3) $\rho \mathcal{R}_H \sigma$ implies $\rho|_K \mathcal{R}_K \sigma|_K$ for all type contexts H and K such that K is a subcontext of H .

Both properties (SHR1) and (GHR1) are called the *substitution* property. Properties (GHR2) and (GHR3) are respectively called the *pre-image* and the *subcontext* properties.

Simple examples of generalized homomorphic relations are easy to construct. The following rather complicated example will be used later to illustrate some important concepts.

Example 2.8. Recall the $\Sigma_{\mathbf{P0}}$ -algebras \mathbf{INT} and \mathbf{NAT} from Exs. 1.7 and 1.8.

Let $H \in TCON$ and $\langle \rho, \sigma \rangle \in ENV_H^{\mathbf{INT} \times \mathbf{NAT}}$ be given. We say that $\langle \rho, \sigma \rangle$ is a *finite partial order isomorphism* if the following holds: for all $x, y \in Dom(H)$ such that $H \vdash x : \mathbf{P0}$ and $H \vdash y : \mathbf{P0}$, $\rho(x) \leq \rho(y)$ iff $\sigma(x) \leq \sigma(y)$. By a *pre-image* of $\langle \rho, \sigma \rangle$ we mean a pair of K -environments of the form $\langle \rho \circ h, \sigma \circ h \rangle$ for some context homomorphism $h : K \rightarrow H$. It is clear that the set of all finite partial isomorphisms is closed under the formation of pre-images.

Let $\mathcal{G}^{\mathbf{P0}}$ be the generalized relation between \mathbf{INT} and \mathbf{NAT} such that

$$\begin{aligned} \mathcal{G}_H^{\mathbf{P0}} := \{ \langle \rho, \sigma \rangle : \langle \rho, \sigma \rangle \text{ is a finite partial order isomorphism} \\ \text{and } \rho(x) = \sigma(x) \text{ whenever } H \vdash x : \mathbf{Bool} \} \end{aligned}$$

Then \mathcal{G}^{P0} is a generalized homomorphic relation.

Proof. To show that \mathcal{G}^{P0} has property (GHR1), let H be a type context and let $\vec{x} \in \text{Dom}(H)^n$ and $\vec{T} \in \text{TYPE}^n$ such that $H \vdash \vec{x} : \vec{T}$. Let $g \in \text{OP}_n$ and let type $\vec{T} \rightarrow S$ be a type of g . Let $y \in \text{VAR} \setminus \text{Dom}(H)$, and let ρ and σ be any pair of H -environments in **INT** and **NAT**, respectively, such that $\rho \mathcal{G}_H^{\text{P0}} \sigma$. If g is a Boolean operation, the conclusion of (2.1) is obvious, so suppose g is **leq**. Let $\text{leq}^{\text{INT}}(\rho(x_1), \rho(x_2)) = a$, and $\text{leq}^{\text{NAT}}(\sigma(x_1), \rho(x_2)) = b$. If $a = tt$, then $\rho(x_1) \leq \rho(x_2)$, and so by the definition of a finite partial order isomorphism, $\sigma(x_1) \leq \sigma(x_2)$, and thus $b = tt$. Similarly, if $a = ff$, then $b = ff$. So by definition of $\mathcal{G}_{H,y:\text{Bool}}^{\text{P0}}$,

$$[y \mapsto a] \rho \mathcal{G}_{H,y:\text{Bool}}^{\text{P0}} [y \mapsto b] \sigma.$$

To show that \mathcal{G}^{P0} has property (GHR2), let type contexts H and K and a context homomorphism h from K to H be given. Suppose $\rho \mathcal{G}_H^{\text{P0}} \sigma$. By the definition of $\mathcal{G}_K^{\text{P0}}$, we must show that: (i) the pair $\langle \rho \circ h, \sigma \circ h \rangle$ is a finite partial order isomorphism, i.e., for all $x, y \in \text{Dom}(K)$ such that $K \vdash x : \text{P0}$ and $K \vdash y : \text{P0}$, $\rho(h(x)) \leq \rho(h(y))$ iff $\sigma(h(x)) \leq \sigma(h(y))$; (ii) $(\rho \circ h)(x) = (\sigma \circ h)(x)$ whenever $K \vdash x : \text{Bool}$, i.e., $\rho(h(x)) = \sigma(h(x))$ whenever $K \vdash x : \text{Bool}$. But (i) follows immediately from the assumption that $\langle \rho, \sigma \rangle$ is a finite partial order isomorphism, and (ii) follows from the assumption that $\rho(x) = \sigma(x)$ whenever $H \vdash x : \text{Bool}$. \square

\mathcal{G}^{P0} is the union of pointwise extensions of standard homomorphic relations. To see this, consider any $\langle \rho, \sigma \rangle \in \mathcal{G}_H^{\text{P0}}$. It is easy to check that $\{ \langle \rho(x), \sigma(x) \rangle : x \in \text{Dom}(H) \}$ is a standard homomorphic relation whose pointwise extension consists of all pre-images of $\langle \rho, \sigma \rangle$. Since \mathcal{G}^{P0} is closed under the formation of pre-images, it must coincide with the union of all the point-wise extensions of standard homomorphic relations associated in the above way with its members.

It turns out that every generalized homomorphic relation is the union of pointwise extensions of standard homomorphic relations (see Thm. A.4 in the Appendix). It is not the case however that every generalized homomorphic relation is the pointwise extension of a single standard homomorphic relation; in fact we shall also see in the Appendix (Ex. A.5) that \mathcal{G}^{P0} itself is not of this form. This and other relationships between standard and generalized homomorphic relations will be studied in the Appendix

According to the substitution property, homomorphic relations are preserved in some sense by the operations of an algebra. The following lemma, whose proof is straightforward, shows that this extends to the process of forming the meanings of terms.

Lemma 2.9. *Let \mathcal{G} be a generalized homomorphic relation between \mathbf{A} and \mathbf{B} . Then for every type context H , for all $\langle \rho, \sigma \rangle \in \text{ENV}^{\mathbf{A} \times \mathbf{B}}$, for all $y \in \text{VAR} \setminus \text{Dom}(H)$, if $\rho \mathcal{G}_H \sigma$, then*

$$[y \mapsto \llbracket t : S \rrbracket^{\mathbf{A}} \rho] \mathcal{G}_{H,y:S} [y \mapsto \llbracket t : S \rrbracket^{\mathbf{B}} \sigma]. \quad \square$$

Corollary 2.10. *Let \mathcal{G} be a generalized homomorphic relation between \mathbf{A} and \mathbf{B} . Then for every type context H , every $\langle \rho, \sigma \rangle \in \text{ENV}^{\mathbf{A} \times \mathbf{B}}$, and for every well H -typed term $t : S$, if $\rho \mathcal{G}_H \sigma$, then $\llbracket t : S \rrbracket^{\mathbf{A}} \rho \mathcal{G}_S^- \llbracket t : S \rrbracket^{\mathbf{B}} \sigma$. \square*

Simulation.

Simulation is naturally homomorphic. If the action of \mathbf{A} is to simulate that of \mathbf{B} , then elements of \mathbf{B} have to be correlated with elements of \mathbf{A} in such a way that if b_0, \dots, b_{n-1} are correlated respectively with a_0, \dots, a_{n-1} , then the action of \mathbf{B} on the b_0, \dots, b_{n-1} , via a $g \in OP_n$ for instance, must be the same that of \mathbf{A} on a_0, \dots, a_{n-1} via the same g . This means that $g^{\mathbf{B}}(b_0, \dots, b_{n-1})$ must be correlated with $g^{\mathbf{A}}(a_0, \dots, a_{n-1})$. Note that simulation does not depend on particular pairs b_i and a_i considered in isolation from other pairs, but on the way \mathbf{B} and \mathbf{A} act on other elements with regard to a_i and b_i . For the notion of simulation to be reasonable, we require that every visible data element simulate only itself. If we are interested only in how visible data behave, this weak notion of simulation is sufficient. But if we are also concerned about the behavior of nonvisible data, then a stronger notion of simulation is necessary. We now make the necessary definitions.

Definition 2.11 (VIS-identical relations). Let $\mathcal{R} \subseteq A \times B$ and $\mathcal{G} \subseteq ENV^{\mathbf{A} \times \mathbf{B}}$.

- (i) \mathcal{R} is *VIS-identical* if $\mathcal{R}|_{VIS} = \mathcal{I}$.
- (ii) \mathcal{G} is *VIS-identical* if $\mathcal{G}|_{VIS} = \mathcal{I}^+$. \square

Example 2.12. The generalized homomorphic relation \mathcal{G}^{P0} of Ex. 2.8, is *VIS-identical*, because when restricted to environments that have only variables of type `Bool`, it is the identity relation. \square

Definition 2.13 (standard simulation). By a *standard simulation of \mathbf{B} by \mathbf{A}* we mean a relation $\mathcal{R} \subseteq A \times B$ that is *VIS-identical*, homomorphic and satisfies the following condition:

- (i) for every $T \in TYPE$ and $a \in A_T$, there exists a $b \in B_T$ such that $a \mathcal{R}_T b$.

\mathcal{R} is a *standard bisimulation between \mathbf{A} and \mathbf{B}* if both \mathcal{R} and its converse, $\check{\mathcal{R}}$, are standard simulations. \square

We say that a *simulates b under \mathcal{R}* if $\langle a, b \rangle \in \mathcal{R}$. Note that the requirement that \mathcal{R} be *VIS-identical* ($\mathcal{R}|_{VIS} = \mathcal{I}$) means that each visible element of \mathbf{A} simulates itself and only itself in \mathbf{B} . We obtain a weaker notion of simulation by requiring only this property and omitting the condition (i) in Def. 2.13; this weaker notion is inherently symmetric and thus gives only a bisimulation.

Definition 2.14 (weak standard bisimulation). By a *weak standard bisimulation between \mathbf{A} and \mathbf{B}* we mean a relation $\mathcal{R} \subseteq A \times B$ that is *VIS-identical* and homomorphic. \square

In a generalized simulation whole environments simulate environments. This leads to a more powerful notion of simulation by taking the contexts in which elements appear into account.

Definition 2.15 (generalized simulation). By a *generalized simulation of \mathbf{B} by \mathbf{A}* we mean a relation $\mathcal{G} \subseteq ENV^{\mathbf{A} \times \mathbf{B}}$ that is *VIS-identical*, homomorphic, and such that for every $H \in TCON$ and $\rho \in ENV_H^{\mathbf{A}}$, there exists a $\sigma \in ENV_H^{\mathbf{B}}$ such that $\rho \mathcal{G}_H \sigma$.

\mathcal{G} is a *generalized bisimulation* between \mathbf{A} and \mathbf{B} if both \mathcal{G} and its converse, $\check{\mathcal{G}}$, are generalized simulations. \square

We will see below in Thm. 4.3 that, if a standard relation \mathcal{R} is homomorphic, then so is its pointwise extension \mathcal{R}^+ . It follows easily from the definitions involved that, if \mathcal{R} is a standard simulation of \mathbf{B} by \mathbf{A} , then \mathcal{R}^+ must be a generalized simulation of \mathbf{B} by \mathbf{A} . So standard (bi)simulation is at least as strong a notion as generalized (bi)simulation. We shall see shortly that it is in fact strictly stronger.

Example 2.16. The generalized relation $\mathcal{G}^{\mathbf{P0}}$ from Ex. 2.8 is a generalized bisimulation between \mathbf{INT} and \mathbf{NAT} .

Proof. We have already seen that it is *VIS*-identical and homomorphic. To show that it is a generalized simulation of \mathbf{NAT} by \mathbf{INT} , we must show that, for every $H \in TCON$ and $\rho \in ENV_H^{\mathbf{INT}}$, there is a $\sigma \in ENV_H^{\mathbf{NAT}}$ such that, in particular, the pair $\langle \rho, \sigma \rangle$ is a finite partial isomorphism. Let x_0, \dots, x_{n-1} be an arrangement of the variables in $Dom(H)$ such that $\rho(x_0) \leq \rho(x_1) \leq \dots \leq \rho(x_{n-1})$. If $\rho(x_0)$, and hence all the $\rho(x_i)$, are nonnegative, then we can take $\sigma = \rho$; otherwise, we can shift each $\rho(x_i)$ to the right by the same amount $|\rho(x_0)|$. The formal definition of σ is as follows. Let $m := \max\{|\rho(x)| : H \vdash x : \mathbf{P0}, \rho(x) \leq 0\}$. Define $\sigma \in ENV_H^{\mathbf{NAT}}$ such that:

$$\begin{aligned} \sigma(x) &= \rho(x) + m, & \text{if } H \vdash x : \mathbf{P0} \\ \sigma(x) &= \rho(x), & \text{if } H \vdash x : \mathbf{Bool}. \end{aligned}$$

Then by definition $\rho \mathcal{G}_H^{\mathbf{P0}} \sigma$. It is even easier to see that the converse of $\mathcal{G}^{\mathbf{P0}}$ is a generalized simulation of \mathbf{INT} by \mathbf{NAT} . \square

We shall see in the Appendix (Ex. A.1) that there is no standard bisimulation between \mathbf{INT} and \mathbf{NAT} . Thus standard bisimulation is strictly stronger than generalized bisimulation.

3 BEHAVIOR AND REALIZATION

In this section we formalize the central notions of behavior and realization. They are similar to notions considered by Goguen and Mesequer in [5,6], and we follow the latter's terminology.

In model theory, a specification can be formalized as an observation together with an expected result. The free variables of such an observation constitute its parameters. For example, one might specify that $\mathbf{1eq}(x, x)$ should have tt as its expected result. An environment ρ over a $\mathbf{P0}$ -algebra \mathbf{NAT} “realizes” this specification if the value of $\llbracket \mathbf{1eq}(x, x) : \mathbf{Bool} \rrbracket^{\mathbf{NAT}} \rho = tt$.

Conversely, one can ask about the behavior of implementations, i.e., about the behavior of environments over algebras. A “behavior” of an environment is the meaning of some observation in that environment. For example, tt is a behavior of ρ , for the observation $\mathbf{1eq}(x, x)$, if $\llbracket \mathbf{1eq}(x, x) : \mathbf{Bool} \rrbracket^{\mathbf{NAT}} \rho = tt$.

Since we are concerned with the behavior of nonvisible data, it is technically simpler to deal with “procedures.” Let H be a type context; formally, any well H -typed term t is called an *H-procedure*; a procedure need not return a result of

visible type. We reserve the term *observation* for procedures that output visible data. Both may have free variables that are of nonvisible types. Finally, by a *program* we mean an observation whose free variables are all of visible type.

Definition 3.1 (behavior and realization). Let \mathbf{A} be a Σ -algebra, and let $H \in TCON$ and $T \in TYPE$. Let $\rho \in ENV_H^{\mathbf{A}}$, $a \in A_T$, and let t be an H -procedure of type T . Then ρ *realizes* a under t and a is the *behavior of* ρ under t if $\llbracket t : T \rrbracket^{\mathbf{A}} \rho = a$. Also, a is a *visible behavior* of ρ if $a \in A_T$ for some $T \in VIS$. \square

To prove correctness of an ADT implementation, one cannot focus on the behavior of a particular data element but must consider the contexts in which it can be used. Consequently, one way to specify ADTs is to focus on the *behavior-function* of environments, ρ , that is, the function that maps each H -procedure t to the behavior $\llbracket t : T \rrbracket^{\mathbf{A}} \rho$ of ρ under t , and then to specify the family of acceptable functions of this kind, say, by some formal specification language.

Alternatively, in the model-theoretic approach, which we follow, the behavior of ρ in \mathbf{A} is compared with the behavior of a paradigm environment σ in some paradigm \mathbf{B} (selected from some class of such paradigms). We shift the focus therefore from the behavior-function to the *comparative behavior relation*, which is a standard relation between \mathbf{A} and \mathbf{B} that associates, for each H -procedure t , the behavior of ρ under t in \mathbf{A} with the behavior of σ under t in \mathbf{B} .

Definition 3.2 (comparative behavior and realization). Let \mathbf{A} and \mathbf{B} be comparable Σ -algebras.

- (i) Let $H \in TCON$ and $\langle \rho, \sigma \rangle \in ENV_H^{\mathbf{A} \times \mathbf{B}}$. The *comparative behavior of* ρ and σ is the standard relation, $\mathcal{BE}(\rho, \sigma)$, defined by

$$\mathcal{BE}(\rho, \sigma)_T := \{ \langle a, b \rangle : \text{for some } H\text{-procedure } t \text{ of type } T, \\ a = \llbracket t : T \rrbracket^{\mathbf{A}} \rho \text{ and } b = \llbracket t : T \rrbracket^{\mathbf{B}} \sigma \}$$

- (ii) Let $T \in TYPE$ and $\langle a, b \rangle \in (A \times B)_T$. The *comparative realization of* a and b is the generalized relation, $\mathcal{RE}(a, b)$, defined by

$$\mathcal{RE}(a, b)_H := \{ \langle \rho, \sigma \rangle : \text{for some } H\text{-procedure } t \text{ of type } T, \\ \llbracket t : T \rrbracket^{\mathbf{A}} \rho = a \text{ and } \llbracket t : T \rrbracket^{\mathbf{B}} \sigma = b \}. \quad \square$$

Note that for all $\langle a, b \rangle \in (A \times B)_T$ and $\langle \rho, \sigma \rangle \in ENV_H^{\mathbf{A} \times \mathbf{B}}$,

$$(3.1) \quad \langle \rho, \sigma \rangle \in \mathcal{RE}(a, b)_H \quad \text{iff} \quad \langle a, b \rangle \in \mathcal{BE}(\rho, \sigma)_T.$$

Example 3.3. Recall the $\Sigma_{\mathbf{P0}}$ -algebras \mathbf{INT} and \mathbf{NAT} from Examples 1.7 and 1.8. Let $H = \{x : \mathbf{P0}, y : \mathbf{P0}\}$, and let the H -environments $\rho \in ENV_H^{\mathbf{INT}}$ and $\sigma \in ENV_H^{\mathbf{NAT}}$ be defined by $\rho = \{\langle x, -3 \rangle, \langle y, 5 \rangle\}$ and $\sigma = \{\langle x, 7 \rangle, \langle y, 8 \rangle\}$. Note that $\langle \rho, \sigma \rangle$ is a finite partial isomorphism as defined in Ex. 2.8. So by definition, $\langle \rho, \sigma \rangle \in \mathcal{G}^{\mathbf{P0}}$, where $\mathcal{G}^{\mathbf{P0}}$ is the generalized homomorphic relation between \mathbf{INT} and \mathbf{NAT} from Ex. 2.8.

The only H -procedures of type $\mathsf{P0}$ are x and y , and the only H -procedures of type Bool are true , false , $\mathsf{leq}(x, x)$, $\mathsf{leq}(x, y)$, $\mathsf{leq}(y, x)$, and $\mathsf{leq}(y, y)$. Hence $\mathcal{BE}(\rho, \sigma)_{\mathsf{P0}} = \{\langle -3, 7 \rangle, \langle 5, 8 \rangle\}$, and $\mathcal{BE}(\rho, \sigma)_{\mathsf{Boo1}}$ is the identity relation on the Booleans, i.e. $\mathcal{BE}(\rho, \sigma)_{\mathsf{Boo1}} = \mathcal{I}$, where \mathcal{I} is the visible standard identity relation between INT and NAT (see Def. 2.2). $\mathcal{RE}(-3, 7)_H$ is the set of all pairs $\langle \rho, \sigma \rangle \in \mathit{ENV}_H^{\mathbf{A} \times \mathbf{B}}$ such that either $\rho(x) = -3$ and $\sigma(x) = 7$ or $\rho(y) = -3$ and $\sigma(y) = 7$. $\mathcal{RE}(tt, tt)_H = \mathit{ENV}_H^{\mathbf{A} \times \mathbf{B}}$.

For future reference, note that $\mathcal{G}^{\mathsf{P0}}$ consists precisely of those pairs of environments $\langle \rho, \sigma \rangle$ such that $\mathcal{BE}(\rho, \sigma)_{\mathsf{Boo1}} = \mathcal{I}$. \square

In the sequel we often speak simply of the *behavior* of a pair of environments instead of their comparative behavior.

The following definition extends the notion of comparative behavior to a family of pairs of environments in the natural way; that is, it associates a standard relation with each generalized relation between \mathbf{A} and \mathbf{B} . Comparative realization can be similarly extended, but we consider the dual notion instead, which turns out to be much more useful for our purposes.

Definition 3.4 (behavior and dual realization operators). Let \mathbf{A} and \mathbf{B} be comparable Σ -algebras.

- (i) Let $\mathcal{G} \subseteq \mathit{ENV}^{\mathbf{A} \times \mathbf{B}}$. Define $\mathcal{BE}(\mathcal{G}) := \langle \mathcal{BE}(\mathcal{G})_T : T \in \mathit{TYPE} \rangle$, where

$$\langle a, b \rangle \in \mathcal{BE}(\mathcal{G})_T \quad \text{iff} \quad \exists H \in \mathit{TCON} \exists \langle \rho, \sigma \rangle \in \mathcal{G}_H (\langle a, b \rangle \in \mathcal{BE}(\rho, \sigma)_T).$$

$\mathcal{BE}(\mathcal{G})$ is called the *behavior* of \mathcal{G} and $\mathcal{BE}(\mathcal{G})|_{\mathit{VIS}}$ is the *visible behavior* of \mathcal{G} . \mathcal{BE} as a function from the generalized to standard relations between \mathbf{A} and \mathbf{B} is called the *behavior operator* on $\mathbf{A} \times \mathbf{B}$.

- (ii) Let $\mathcal{R} \subseteq \mathbf{A} \times \mathbf{B}$. Define $\mathcal{RE}^\partial(\mathcal{R}) := \langle \mathcal{RE}^\partial(\mathcal{R})_H : H \in \mathit{TCON} \rangle$, where

$$\langle \rho, \sigma \rangle \in \mathcal{RE}^\partial(\mathcal{R})_H \quad \text{iff} \quad \forall T \in \mathit{TYPE} \forall \langle a, b \rangle \notin \mathcal{R}_T (\langle \rho, \sigma \rangle \notin \mathcal{RE}(a, b)_H).$$

$\mathcal{RE}^\partial(\mathcal{R})$ is called the *dual realization* of \mathcal{R} . \mathcal{RE}^∂ as a function from the standard to generalized relations between \mathbf{A} and \mathbf{B} is called the *dual realization operator* on $\mathbf{A} \times \mathbf{B}$. \square

Note that for all $H \in \mathit{TCON}$ and $\langle \rho, \sigma \rangle \in \mathit{ENV}_H^{\mathbf{A} \times \mathbf{B}}$,

$$(3.2) \quad \langle \rho, \sigma \rangle \in \mathcal{RE}^\partial(\mathcal{R})_H \quad \text{iff} \quad \forall T \in \mathit{TYPE} \mathcal{BE}(\rho, \sigma)_T \subseteq \mathcal{R}_T.$$

Example 3.5. Let $\mathcal{G}^{\mathsf{P0}}$ be the generalized relation of Ex. 2.8. As we have already observed in Ex. 3.3, $\mathcal{BE}(\mathcal{G}^{\mathsf{P0}})_{\mathsf{Boo1}}$ is the identity relation, and it is easy to see that $\mathcal{BE}(\mathcal{G}^{\mathsf{P0}})_{\mathsf{P0}} = \mathbb{Z} \times \mathbb{N}$. \square

Before giving an example of the dual realization operator, we define the notion of an extended visible identity relation. The standard relation $\mathcal{BE}(\mathcal{G}^{\mathsf{P0}})$, which as we have observed above is the identity relation on the visible type of Σ_{P0} and the universal relation on the nonvisible type, is a special case of general class of extended visible identity relations that proves to be quite useful in the sequel.

Definition 3.6 (extended visible standard identity relation). For each pair of comparable algebras \mathbf{A} and \mathbf{B} , the *extended visible standard identity relation* \mathcal{I}^* is defined by

$$\mathcal{I}_T^* = \begin{cases} \mathcal{I}_T (= \{ \langle a, a \rangle : a \in A_T \}), & \text{if } T \in \text{VIS} \\ A_T \times B_T, & \text{if } T \in \text{TYPE} \setminus \text{VIS}. \quad \square \end{cases}$$

Example 3.7. Consider the extended visible standard identity relation \mathcal{I}^* between **INT** and **NAT**. Its dual realization, $\mathcal{RE}^\partial(\mathcal{I}^*)$ is the generalized relation \mathcal{G}^{PO} of Example 2.8. This follows easily from (3.2) and the observation made in Ex. 3.5. \square

The next in the next Theorem collects the basic properties of the behavior and dual realizations operators. Its proof is straightforward.

Theorem 3.8. For all $\mathcal{R}, \mathcal{S} \subseteq A \times B$ and all $\mathcal{G}, \mathcal{H} \subseteq \text{ENV}^{A \times B}$:

- (i) $\mathcal{G} \subseteq \mathcal{H}$ implies $\mathcal{BE}(\mathcal{G}) \subseteq \mathcal{BE}(\mathcal{H})$;
- (ii) $\mathcal{R} \subseteq \mathcal{S}$ implies $\mathcal{RE}^\partial(\mathcal{R}) \subseteq \mathcal{RE}^\partial(\mathcal{S})$;
- (iii) $\mathcal{BE}(\mathcal{RE}^\partial(\mathcal{R})) \subseteq \mathcal{R}$;
- (iv) $\mathcal{G} \subseteq \mathcal{RE}^\partial(\mathcal{BE}(\mathcal{G}))$. \square

The sets of standard and generalized relations between fixed \mathbf{A} and \mathbf{B} are partially ordered sets (posets) by set-theoretical inclusion, and the operators \mathcal{BE} and \mathcal{RE}^∂ are mappings between these two posets. Thm. 3.8 says that \mathcal{BE} and \mathcal{RE}^∂ form a Galois connection when viewed as mappings between the poset of standard relations and the dual poset of generalized relations (see e.g. Birkhoff [2], p.124).

The basic adjunction. Like all posets, the standard and generalized relations between \mathbf{A} and \mathbf{B} can be viewed as *simple* categories, i.e., categories in which there is at most one arrow between any pair of objects. \mathcal{BE} and \mathcal{RE}^∂ and their duals preserve inclusion and thus are functors between the two categories. In the following corollary we give the well-known alternative characterization of the Galois connection as an adjunction between simple categories. We will use this adjunction repeatedly in the sequel.

Corollary 3.9 (Basic Adjunction). Then for every $\mathcal{R} \subseteq A \times B$ and every $\mathcal{G} \subseteq \text{ENV}^{A \times B}$:

$$\mathcal{BE}(\mathcal{G}) \subseteq \mathcal{R} \quad \text{iff} \quad \mathcal{G} \subseteq \mathcal{RE}^\partial(\mathcal{R}).$$

Proof. If $\mathcal{BE}(\mathcal{G}) \subseteq \mathcal{R}$, then by Thm. 3.8(ii)(iv), $\mathcal{G} \subseteq \mathcal{RE}^\partial(\mathcal{BE}(\mathcal{G})) \subseteq \mathcal{RE}^\partial(\mathcal{R})$. Conversely, if $\mathcal{G} \subseteq \mathcal{RE}^\partial(\mathcal{R})$, then by Thm. 3.8(i)(iii), $\mathcal{BE}(\mathcal{G}) \subseteq \mathcal{BE}(\mathcal{RE}^\partial(\mathcal{R})) \subseteq \mathcal{R}$. \square

The basic adjunction can be paraphrased in the following way. For every standard relation \mathcal{R} , its dual realization $\mathcal{RE}^\partial(\mathcal{R})$ is the largest generalized relation whose behavior is included in \mathcal{R} , and for every generalized relation \mathcal{G} , its behavior $\mathcal{BE}(\mathcal{G})$ is the smallest standard relation whose dual realization includes \mathcal{G} .

By specializing to the behaviors of generalized relations of the form $\{ \langle \rho, \sigma \rangle \}$ we get the following useful local version of the basic adjunction.

Corollary 3.10. *Let \mathbf{A}, \mathbf{B} be comparable and $H \in TCON$. Then for every $\langle \rho, \sigma \rangle \in ENV_H^{\mathbf{A} \times \mathbf{B}}$ and every $\mathcal{R} \subseteq A \times B$:*

$$\mathcal{BE}(\rho, \sigma) \subseteq \mathcal{R} \quad \text{iff} \quad \rho \mathcal{RE}^\partial(\mathcal{R})_H \sigma. \quad \square$$

We now identify certain relations as behaviors and dual realizations.

Definition 3.11 (behavior and dual realization).

- (i) $\mathcal{R} \subseteq A \times B$ is a *behavior* if $\mathcal{BE}(\mathcal{RE}^\partial(\mathcal{R})) = \mathcal{R}$.
- (ii) $\mathcal{G} \subseteq ENV^{\mathbf{A} \times \mathbf{B}}$ is a *dual realization* if $\mathcal{RE}^\partial(\mathcal{BE}(\mathcal{G})) = \mathcal{G}$. \square

It follows from the fact that \mathcal{BE} and \mathcal{RE}^∂ form a Galois connection that \mathcal{R} is a behavior iff $\mathcal{R} = \mathcal{BE}(\mathcal{G})$ for some $\mathcal{G} \subseteq ENV^{\mathbf{A} \times \mathbf{B}}$ and that \mathcal{G} is a dual realization iff $\mathcal{G} = \mathcal{RE}^\partial(\mathcal{R})$ for some $\mathcal{R} \subseteq A \times B$. The sets of behaviors and dual realizations form isomorphic complete lattices under set-theoretic inclusion.

Lemma 3.12. *Let \mathbf{A} and \mathbf{B} be comparable and let \mathcal{I} be the visible standard identity relation between them. Then $\mathcal{I} \subseteq \mathcal{R}$ for every nonempty behavior \mathcal{R} . It follows immediately that*

- (i) for every nonempty $\mathcal{G} \subseteq ENV^{\mathbf{A} \times \mathbf{B}}$, $\mathcal{I} \subseteq \mathcal{BE}(\mathcal{G})$;
- (ii) for every $H \in TCON$ and $\langle \rho, \sigma \rangle \in ENV_H^{\mathbf{A} \times \mathbf{B}}$, $\mathcal{I} \subseteq \mathcal{BE}(\rho, \sigma)$.

Proof. Let \mathcal{R} be any nonempty behavior. Then $\mathcal{R} = \mathcal{BE}(\mathcal{G})$ for some nonempty \mathcal{G} (\mathcal{G} is nonempty since the empty generalized relation has empty behavior). Let $H \in TCON$ be such that \mathcal{G}_H is nonempty and let ρ, σ be H -environments such that $\rho \mathcal{G}_H \sigma$. Every ground term $t : T$ is an H -procedure. Thus $t^{\mathbf{A}} = \llbracket t : T \rrbracket^{\mathbf{A}} \rho \mathcal{BE}(\mathcal{G})_T$ $\llbracket t : T \rrbracket^{\mathbf{B}} \sigma = t^{\mathbf{B}}$. Since $\mathbf{A}|_{VIS} = \mathbf{B}|_{VIS}$ is term-generated, every visible data element is the value of a ground term. Hence, $\mathcal{I} \subseteq \mathcal{BE}(\mathcal{G})$. \square

Recall that the extended visible standard identity relation \mathcal{I}^* between \mathbf{A} and \mathbf{B} is the identity on visible types and the universal relation on nonvisible types (Def. 3.6). \mathcal{I}^* is useful because of the following property:

$$(3.3) \quad \mathcal{R}|_{VIS} \subseteq \mathcal{I} \quad \text{iff} \quad \mathcal{R} \subseteq \mathcal{I}^*, \quad \text{for every } \mathcal{R} \subseteq A \times B.$$

We have the following consequence of the basic adjunction and its local version. Recall that $\mathcal{R} \subseteq A \times B$ is *VIS-identical* if $\mathcal{R}|_{VIS} = \mathcal{I}$ (Def. 2.11).

Corollary 3.13.

- (i) $\mathcal{RE}^\partial(\mathcal{I}^*)$ is the largest generalized relation between \mathbf{A} and \mathbf{B} whose behavior is *VIS-identical*, i.e.,

$$\text{for all nonempty } \mathcal{G} \subseteq ENV^{\mathbf{A} \times \mathbf{B}}, \quad \mathcal{BE}(\mathcal{G})|_{VIS} = \mathcal{I} \quad \text{iff} \quad \mathcal{G} \subseteq \mathcal{RE}^\partial(\mathcal{I}^*).$$

- (ii) Let $H \in TCON$.

$$\text{for all } \langle \rho, \sigma \rangle \in ENV_H^{\mathbf{A} \times \mathbf{B}}, \quad \mathcal{BE}(\rho, \sigma)|_{VIS} = \mathcal{I} \quad \text{iff} \quad \rho \mathcal{RE}^\partial(\mathcal{I}^*) \sigma.$$

Proof. (i). Let $\mathcal{G} \subseteq ENV^{\mathbf{A} \times \mathbf{B}}$ with $\mathcal{G} \neq \emptyset$. Since $\mathcal{I} \subseteq \mathcal{BE}(\mathcal{G})|_{VIS}$ by Lem. 3.12, we have $\mathcal{BE}(\mathcal{G})|_{VIS} = \mathcal{I}$ iff $\mathcal{BE}(\mathcal{G})|_{VIS} \subseteq \mathcal{I}$. Now applying first the equivalence (3.3) and then the basic adjunction we get

$$\mathcal{BE}(\mathcal{G})|_{VIS} = \mathcal{I} \quad \text{iff} \quad \mathcal{BE}(\mathcal{G}) \subseteq \mathcal{I}^* \quad \text{iff} \quad \mathcal{G} \subseteq \mathcal{RE}^\partial(\mathcal{I}^*).$$

(ii) follows similarly from the local version of the basic adjunction. \square

VIS-behavioral reducibility and equivalence. We now apply this machinery to visible behavior. In most practical situations one is interested in the visible behavior of H -environments, that is, the function that assigns to each (visible) H -observation $t : V$ the value $\llbracket t : V \rrbracket^{\mathbf{A}} \rho$. We refine the notion of comparable behavior accordingly.

Definition 3.14 (VIS-behavioral equivalence). Let $H \in TCON$ and $\langle \rho, \sigma \rangle \in ENV_H^{\mathbf{A} \times \mathbf{B}}$. Then ρ and σ are *VIS-behaviorally equivalent* iff $\mathcal{BE}(\rho, \sigma)$ is *VIS-identical*, i.e., $\mathcal{BE}(\rho, \sigma)|_{VIS} = \mathcal{I}$. \square

By Cor. 3.13, ρ and σ are *VIS-behaviorally equivalent* iff $\rho \mathcal{RE}^\partial(\mathcal{I}^*) \sigma$.

According to Schoett [18], comparable algebras \mathbf{A} and \mathbf{B} are behaviorally equivalent if every visible environment is *VIS-behaviorally equivalent* to itself when viewed as an environment of \mathbf{A} and then of \mathbf{B} ; that is, if for every visible type context H (see Def. 1.4) the following holds for every H -environment ρ of $\mathbf{A}|_{VIS} = \mathbf{B}|_{VIS}$:

$$\llbracket t : V \rrbracket^{\mathbf{A}} \rho = \llbracket t : V \rrbracket^{\mathbf{B}} \rho, \quad \text{every } H\text{-observation } t : V.$$

We consider a stronger notion of behavioral equivalence that takes into account all environments, not just visible ones. Since there is no reasonable way to identify the nonvisible environments of \mathbf{A} with those of \mathbf{B} , we first define the asymmetric notion of behavioral reducibility. We say that \mathbf{A} is *VIS-behaviorally reducible* to \mathbf{B} if, for every environment ρ in \mathbf{A} (visible or nonvisible), we can find an environment σ in \mathbf{B} that is *VIS-behaviorally equivalent* to it. Furthermore, if ρ is visible, σ must equal ρ ; i.e., ρ must have the same visible behavior in both algebras. Recall the definition of the visible generalized identity relation given in Def. 2.2.

Definition 3.15 (VIS-behavioral reducibility and equivalence). Let \mathbf{A} and \mathbf{B} be comparable algebras. \mathbf{A} is *VIS-behaviorally reducible* to \mathbf{B} if both the following conditions hold:

- (i) for every $H \in TCON$ and $\rho \in ENV_H^{\mathbf{A}}$, there exists a $\sigma \in ENV_H^{\mathbf{B}}$ such that $\mathcal{BE}(\rho, \sigma)|_{VIS} = \mathcal{I}$, or equivalently, $\rho \mathcal{RE}^\partial(\mathcal{I}^*)_H \sigma$;
- (ii) the behavior of \mathcal{I}^+ is *VIS-identical*, i.e., $\mathcal{BE}(\mathcal{I}^+)|_{VIS} = \mathcal{I}$.

The Σ -algebras \mathbf{A} and \mathbf{B} are *VIS-behaviorally equivalent* if each of \mathbf{A} and \mathbf{B} is *VIS-behaviorally reducible* to the other. \square

The equivalence of the two conditions in part (i) is Cor. 3.13. We note that the condition (ii) of Def. 3.15 is essentially identical to Schoett's weaker notion of behavioral equivalence. We formalize it as follows.

Definition 3.16 (weak VIS-behavioral equivalence [18]). Algebras \mathbf{A} and \mathbf{B} are *weakly VIS-behaviorally equivalent* iff the behavior of \mathcal{I}^+ is VIS-identical, i.e., $\mathcal{BE}(\mathcal{I}^+) |_{VIS} = \mathcal{I}$. \square

Weak VIS-behavioral equivalence refers only to the behavior of visible environments. It is not difficult to find examples of comparable algebras that are not weakly VIS-behaviorally equivalent. In fact, since the visible parts of all algebras are assumed to be term-generated, it is not hard to see that \mathbf{A} and \mathbf{B} fail to be weakly VIS-behaviorally equivalent iff there exists a ground Σ -term t of visible type such that $t^{\mathbf{A}} \neq t^{\mathbf{B}}$. Note however, that if \mathbf{A} and \mathbf{B} are both models of a sufficiently complete specification ([9]), then by definition every ground visible Σ -term is logically equivalent to some ground Σ_{VIS} -term, and hence \mathbf{A} and \mathbf{B} are automatically weakly VIS-behaviorally equivalent.

We now turn to the study of a stronger notion of VIS-behavioral equivalence.

Proposition 3.17. \mathbf{A} and \mathbf{B} are VIS-behaviorally equivalent iff the dual realization of \mathcal{I}^* is VIS-identical, i.e., $\mathcal{I}^+ = \mathcal{RE}^\partial(\mathcal{I}^*) |_{VIS}$.

Proof. Applying the adjunction of Cor. 3.13(i) with $\mathcal{G} = \mathcal{I}^+$, we get $\mathcal{BE}(\mathcal{I}^+) |_{VIS} = \mathcal{I}$ iff $\mathcal{I}^+ \subseteq \mathcal{RE}^\partial(\mathcal{I}^*)$. Thus \mathbf{A} and \mathbf{B} are weakly VIS-behaviorally equivalent iff $\mathcal{I}^+ \subseteq \mathcal{RE}^\partial(\mathcal{I}^*) |_{VIS}$. It remains to verify the inclusion $\mathcal{RE}^\partial(\mathcal{I}^*) |_{VIS} \subseteq \mathcal{I}^+$.

Let $H \in TCON |_{VIS}$ and assume $\rho \mathcal{RE}^\partial(\mathcal{I}^*)_H \sigma$. Then by Cor. 3.10 $\mathcal{BE}(\rho, \sigma) \subseteq \mathcal{I}^*$, and hence

$$(3.4) \quad \mathcal{BE}(\rho, \sigma) |_{VIS} \subseteq \mathcal{I}.$$

Consider any $x \in Dom(H)$. Then $H \vdash x : V$ with $V \in VIS$. So $x : V$ is a visible H -observation and hence (3.4) implies $\rho(x) = \llbracket x : T \rrbracket^{\mathbf{A}} \rho = \llbracket x : T \rrbracket^{\mathbf{B}} \sigma = \sigma(x)$. Thus $\rho(x) = \sigma(x)$ for all $x \in Dom(H)$, i.e., $\rho = \sigma$. \square

4 HOMOMORPHIC BEHAVIOR AND DUAL REALIZATION

In this section we answer the following question: when are behavior and dual realization homomorphic, and thus candidates for simulations? It turns out that dual realization is always homomorphic (Thm. 4.10), but that behavior is homomorphic only under certain special circumstances (Cor. 4.4 and Thm. 4.7). These results will then be used to specify the exact correlation between simulation and VIS-behavioral equivalence in Thms. 4.9 and 4.13. As a start towards these results, we show that the projective restriction of a generalized homomorphic relation is always a behavior.

Proposition 4.1. Let $\mathcal{G} \subseteq ENV^{\mathbf{A} \times \mathbf{B}}$. If \mathcal{G} is homomorphic, then $\mathcal{G}^- = \mathcal{BE}(\mathcal{G})$.

Proof. We first show that, for any generalized relation \mathcal{G} , $\mathcal{G}^- \subseteq \mathcal{BE}(\mathcal{G})$. Let $T \in TYPE$ and $\langle a, b \rangle \in (A \times B)_T$. Then, by the definition of \mathcal{G}^- , $a \mathcal{G}^- b$ iff there is a $H \in TCON$ and a $\langle \rho, \sigma \rangle \in \mathcal{G}_H$ such that $\langle a, b \rangle = \langle \rho(x), \sigma(x) \rangle$ for some $x \in Dom(H)$ such that $H \vdash x : T$. Applying the definition of $\mathcal{BE}(\mathcal{G})$ to the H -procedure $x : T$, we get that $a \mathcal{G}^- b$ implies $a = \llbracket x : T \rrbracket^{\mathbf{A}} \rho \mathcal{BE}(\mathcal{G})_T \llbracket x : T \rrbracket^{\mathbf{B}} \sigma = b$. Hence $\mathcal{G}^- \subseteq \mathcal{BE}(\mathcal{G})$.

Now assume \mathcal{G} is homomorphic. Let $a \in A_T$ and $b \in B_T$ be given and suppose $a \mathcal{BE}(\mathcal{G})_T b$. Then by definition of $\mathcal{BE}(\mathcal{G})$ there exist $H \in TCON$, an H -procedure $t : T$, and H -environments ρ and σ in \mathbf{A} and \mathbf{B} , respectively, such that $a = \llbracket t : T \rrbracket^{\mathbf{A}} \rho$, $b = \llbracket t : T \rrbracket^{\mathbf{B}} \sigma$, and $\rho \mathcal{G}_H \sigma$. By Cor. 2.10 we have $a \mathcal{G}_T^- b$. \square

Corollary 4.2. *Let $\mathcal{G} \subseteq ENV^{A \times B}$. Assume \mathcal{G} is homomorphic. Then if \mathcal{G} is VIS-identical, so is its behavior. \square*

Theorem 4.3. *Let \mathcal{R} be a standard relation. The following are equivalent.*

- (i) \mathcal{R} is homomorphic;
- (ii) \mathcal{R}^+ is homomorphic;
- (iii) $\mathcal{R} = \mathcal{BE}(\mathcal{R}^+)$.

Proof. (i) implies (ii): Assume \mathcal{R} is homomorphic. Let H be a type context and ρ and σ H -environments such that $\rho \mathcal{R}_H^+ \sigma$. By definition of \mathcal{R}^+ , we have

$$(4.1) \quad \rho(x) \mathcal{R}_{H(x)} \sigma(x), \text{ for all } x \in \text{Dom}(H).$$

To show that \mathcal{R}^+ has the pre-image property, let $K \in TCON$ and let $h: K \rightarrow H$. We must show that $\rho \circ h \mathcal{R}_K^+ \sigma \circ h$. This follows from

$$(\rho \circ h)(y) \mathcal{R}_{K(y)} (\sigma \circ h)(y), \text{ for all } y \in \text{Dom}(K).$$

But this in turn follows immediately from formula (4.1) and the fact that $K(y) = H(h(y))$ and $(\rho \circ h)(y) = \rho(h(y))$.

To verify the substitution property, let $g \in OP_n$ and $\vec{x} \in \text{Dom}(H)^n$. Let $\vec{T} \rightarrow S$ be a type of g such that $H \vdash \vec{x} : \vec{T}$. We must show, for all variables $y \notin \text{Dom}(H)$,

$$(4.2) \quad [y \mapsto g^A(\rho(\vec{x}))] \rho \mathcal{R}_{H, y; S}^+ [y \mapsto g^B(\sigma(\vec{x}))] \sigma.$$

But $\rho(\vec{x}) \mathcal{R}_{\vec{T}} \sigma(\vec{x})$ by formula (4.1), and hence $g^A(\rho(\vec{x})) \mathcal{R}_S g^B(\sigma(\vec{x}))$ since \mathcal{R} is homomorphic. Formula (4.2) now follows immediately from formula (4.1).

(ii) implies (iii): If \mathcal{R}^+ is homomorphic, then $\mathcal{R} = \mathcal{R}^{+-} = \mathcal{BE}(\mathcal{R}^+)$; The first equality holds for any standard relation (Lem. 2.5), and the second follows by Prop. 4.1.

(iii) implies (i): Assume $\mathcal{R} = \mathcal{BE}(\mathcal{R}^+)$. To verify that \mathcal{R} has the substitution property, let $g \in OP_n$ and $\vec{T} \rightarrow S$ be a type of g . Let $\vec{a} \in A^n$, $\vec{b} \in B^n$ be such that $\vec{a} \mathcal{R}_{\vec{T}} \vec{b}$. We must show

$$(4.3) \quad g^A(\vec{a}) \mathcal{R}_S g^B(\vec{b}).$$

Choose any $\vec{x} \in VAR^n$ and let $H := \vec{x} : \vec{T}$. Let ρ and σ be the H -environments such that $\rho(\vec{x}) = \vec{a}$ and $\sigma(\vec{x}) = \vec{b}$. Then $\rho \mathcal{R}^+ \sigma$, and hence

$$g^A(\vec{a}) = \llbracket g(\vec{x}) : S \rrbracket^A \rho \mathcal{BE}(\mathcal{R}^+)_S \llbracket g(\vec{x}) : S \rrbracket^B \sigma = g^B(\vec{b}).$$

This gives formula (4.3). \square

Corollary 4.4. *If \mathcal{R} is homomorphic standard relation, then so is $\mathcal{BE}(\mathcal{R}^+)$. \square*

Homomorphic behavior. The last theorem gives one condition for a generalized relation to have homomorphic behavior. Theorem 4.7 below describes a much larger class with this property. We first note that the comparative behavior of any pair of environments is homomorphic.

Proposition 4.5. *For every $H \in TCON$ and $\langle \rho, \sigma \rangle \in ENV_H^{\mathbf{A} \times \mathbf{B}}$, $\mathcal{BE}(\rho, \sigma)$ is a standard homomorphic relation.*

Proof. Let $g \in OP_n$ and $\vec{T} \rightarrow S$ be a type of g . Suppose $\vec{a} \in A_{\vec{T}}$ and $\vec{b} \in B_{\vec{T}}$ is such that $\vec{a} \mathcal{BE}(\rho, \sigma)_{\vec{T}} \vec{b}$. Then by definition of $\mathcal{BE}(\rho, \sigma)$, there exists, for each i , an H -procedure $t_i : T_i$ such that $\llbracket t_i : T_i \rrbracket^{\mathbf{A}} \rho = a_i$ and $\llbracket t_i : T_i \rrbracket^{\mathbf{B}} \sigma = b_i$. Thus

$$g^{\mathbf{A}}(\vec{a}) = \llbracket g(t_1, \dots, t_n) : S \rrbracket^{\mathbf{A}} \rho \mathcal{BE}(\rho, \sigma)_S \llbracket g(t_1, \dots, t_n) : S \rrbracket^{\mathbf{B}} \sigma = g^{\mathbf{B}}(\vec{b}). \quad \square$$

We now develop the mechanism for isolating the property of an arbitrary generalized relation that will allow us to infer the homomorphic character of its behavior from that of its component pairs of environments.

Let $H = \{x_1 : T_1, \dots, x_n : T_n\}$ and $K = \{y_1 : S_1, \dots, y_m : S_m\}$ be type contexts. By the *disjoint union* of H and K , in symbols $H \sqcup K$, we mean the type context $\{x_1 : T_1, \dots, x_n : T_n, y'_1 : S_1, \dots, y'_m : S_m\}$, where, for each $i = 1, \dots, m$, $y'_i = y_i$ if $y_i \notin \{x_1, \dots, x_n\}$, and otherwise, y'_i is the first variable z (in a fixed standard ordering of the variables) such that $z \notin \{x_1, \dots, x_n, y'_1, \dots, y'_{i-1}\}$. Note that H is a subcontext of $H \sqcup K$ and K is obtained from a subcontext of $H \sqcup K$ by a change of variables.

Let ρ and μ be H - and K -environments, respectively. The *disjoint union* $\rho \sqcup \mu$ of ρ and μ , a $H \sqcup K$ -environment, is defined in the obvious way: $(\rho \sqcup \mu)(x) = \rho(x)$ for each $x \in Dom(H)$ and $(\rho \sqcup \mu)(y') = \mu(y)$ for each $y \in Dom(K)$. The following lemma characterizes the behavior of the disjoint union of pairs of environments.

Lemma 4.6. *Let $H, K \in TCON$, $\langle \rho, \sigma \rangle \in ENV_H^{\mathbf{A} \times \mathbf{B}}$, and $\langle \mu, \nu \rangle \in ENV_K^{\mathbf{A} \times \mathbf{B}}$. Then*

$$\mathcal{BE}(\rho, \sigma) \cup \mathcal{BE}(\mu, \nu) \subseteq \mathcal{BE}(\rho \sqcup \mu, \sigma \sqcup \nu).$$

Proof. Suppose $a \mathcal{BE}(\mu, \nu)_T b$. Then there is an K -procedure $t : T$ such that $a = \llbracket t : T \rrbracket^{\mathbf{A}} \mu$ and $b = \llbracket t : T \rrbracket^{\mathbf{B}} \nu$. Let μ', ν', t' , and T' be obtained respectively from μ, ν, t , and T by the appropriate change of variables. Then $t' : T'$ is a $(H \sqcup K)$ -procedure and $a = \llbracket t' : T' \rrbracket^{\mathbf{A}} \mu'$ and $b = \llbracket t' : T' \rrbracket^{\mathbf{B}} \nu'$. Hence $a \mathcal{BE}(\rho \sqcup \mu, \sigma \sqcup \nu)_T b$. In a similar way we get $\mathcal{BE}(\rho, \sigma) \subseteq \mathcal{BE}(\rho \sqcup \mu, \sigma \sqcup \nu)$. \square

The next theorem says that if a generalized relation preserves disjoint unions, then it has homomorphic behavior.

Theorem 4.7. *Let \mathcal{G} be a generalized relation between \mathbf{A} and \mathbf{B} . Assume that \mathcal{G} is closed under disjoint union in the following sense: for all H, K, ρ, σ, μ , and ν such that $H, K \in TCON$, $\rho \in ENV_H^{\mathbf{A}}$, $\sigma \in ENV_H^{\mathbf{B}}$, $\mu \in ENV_K^{\mathbf{A}}$, and $\nu \in ENV_K^{\mathbf{B}}$, we have that*

$$\rho \mathcal{G}_H \sigma \text{ and } \mu \mathcal{G}_K \nu \text{ imply } (\rho \sqcup \mu) \mathcal{G}_{H \sqcup K} (\sigma \sqcup \nu).$$

Then $\mathcal{BE}(\mathcal{G})$ is homomorphic.

Proof. Let $g \in OP_n$. Let $\vec{T} = T_1 \cdots T_n$ be a type domain of g and S the corresponding result type. Let $\vec{a} \in A_{\vec{T}}$ and $\vec{b} \in B_{\vec{T}}$ such that $\vec{a} \mathcal{BE}(\mathcal{G})_{\vec{T}} \vec{b}$. For each i there is a $H_i \in TCON$ and H_i -environments ρ_i, σ_i such that $\rho_i \mathcal{G}_{H_i} \sigma_i$ and $a_i \mathcal{BE}(\rho_i, \sigma_i)_{T_i} b_i$.

By Lem. 4.6 we have $a_i \mathcal{BE}(\rho_1 \sqcup \dots \sqcup \rho_n, \sigma_1 \sqcup \dots \sqcup \sigma_n)_{T_i} b_i$ for all i . Thus by Prop. 4.5 $g^{\mathbf{A}}(\vec{a}) \mathcal{BE}(\rho_1 \sqcup \dots \sqcup \rho_n, \sigma_1 \sqcup \dots \sqcup \sigma_n)_S g^{\mathbf{B}}(\vec{b})$. But by hypothesis

$$(\rho_1 \sqcup \dots \sqcup \rho_n) \mathcal{G}_{H_1 \sqcup \dots \sqcup H_n} (\sigma_1 \sqcup \dots \sqcup \sigma_n).$$

So by definition $g^{\mathbf{A}}(\vec{a}) \mathcal{BE}(\mathcal{G})_S g^{\mathbf{B}}(\vec{b})$. \square

Corollary 4.8. $\mathcal{BE}(\mathcal{I}^+)$ is homomorphic.

Proof. Let $H, K \in TCON$ and let $\rho \mathcal{I}_H^+ \sigma$ and $\mu \mathcal{I}_K^+ \nu$. Then H and K are both visible and $\rho = \sigma$ and $\mu = \nu$. But then $\rho \sqcup \mu = \sigma \sqcup \nu$ and hence $(\rho \sqcup \mu) \mathcal{I}_{H \sqcup K}^+ (\sigma \sqcup \nu)$. Thus $\mathcal{BE}(\mathcal{I}^+)$ is homomorphic by Thm. 4.7. \square

By the basic adjunction, $\mathcal{BE}(\mathcal{I}^+)$ is the smallest standard homomorphic relation whose dual realization includes \mathcal{I}^+ .

We now have the machinery in place to prove Schoett's algebraic characterization of weak *VIS*-behavioral equivalence (Def. 3.16).

Theorem 4.9 (Schoett [18]). *Algebras \mathbf{A} and \mathbf{B} are weakly *VIS*-behaviorally equivalent iff there exists a weak bisimulation between \mathbf{A} and \mathbf{B} .*

Proof. Assume \mathbf{A} and \mathbf{B} are weakly *VIS*-behaviorally equivalent, i.e., assume that $\mathcal{BE}(\mathcal{I}^+) |_{VIS} = \mathcal{I}$. Since $\mathcal{BE}(\mathcal{I}^+)$ is also homomorphic by Cor 4.8, $\mathcal{BE}(\mathcal{I}^+)$ itself is the desired weak bisimulation (Def. 2.14). Assume now that there exists a weak bisimulation \mathcal{R} between \mathbf{A} and \mathbf{B} . Then \mathcal{R} is *VIS*-identical by definition, i.e., $\mathcal{R} |_{VIS} = \mathcal{I}$. Thus $\mathcal{I} \subseteq \mathcal{R}$ and hence $\mathcal{BE}(\mathcal{I}^+) \subseteq \mathcal{BE}(\mathcal{R}^+)$ since $^+$ and \mathcal{BE} are both monotonic operators. But \mathcal{R} is homomorphic so $\mathcal{BE}(\mathcal{R}^+) = \mathcal{R}$ by Thm. 4.3(iii); hence $\mathcal{BE}(\mathcal{I}^+) |_{VIS} \subseteq \mathcal{R} |_{VIS} = \mathcal{I}$. Thus \mathbf{A} and \mathbf{B} are weakly *VIS*-behaviorally equivalent by definition. \square

Note that, if any homomorphic *VIS*-identical standard relation between \mathbf{A} and \mathbf{B} exists, then $\mathcal{BE}(\mathcal{I}^+)$ is the smallest. Hence checking that $\mathcal{BE}(\mathcal{I}^+)$ is homomorphic is necessary and sufficient for establishing a weak-bisimulation. This might be useful in devising algorithms to perform such checks.

Extension of Schoett's theorem. Schoett's theorem provides an algebraic characterization of weak *VIS*-behavioral equivalence, but it deals only with the behavior of the same visible environment in two different algebras. Although standard relations provide an algebraic characterization of weak *VIS*-behavioral equivalence, we will see in the Appendix that they are incapable of characterizing *VIS*-behavioral equivalence in general. For this purpose one has to be able to compare the behavior of different, nonvisible environments, i.e., one has to turn to generalized relations. Thus we use dual realization as our main tool in our extension of Schoett's theorem. We begin our study of by showing that, in contrast to the situation for behavior, the dual realization of every standard relation is homomorphic.

Theorem 4.10. *Let \mathcal{R} be a standard relation. Then $\mathcal{RE}^\partial(\mathcal{R})$ is a generalized homomorphic relation.*

Proof. We verify that $\mathcal{RE}^\partial(\mathcal{R})$ has the properties (GHR1) and (GHR2). Let H be a type context and let $\vec{y} \in Dom(H)^n$ and $\vec{T} \in TYPE^n$ such that $H \vdash \vec{y} : \vec{T}$. Let

$g \in OP_n$ and $\vec{T} \rightarrow S$ be a type of g . Let $z \in VAR \setminus Dom(H)$ and let ρ and σ be H -environments in \mathbf{A} and \mathbf{B} , respectively. Assume

$$(4.4.) \quad \rho \mathcal{RE}^\partial(\mathcal{R})_H \sigma$$

To verify (GHR1) we must show

$$(4.5) \quad [z \mapsto g^{\mathbf{A}}(\rho(\vec{y}))]\rho \mathcal{RE}^\partial(\mathcal{R})_{H,z:S} [z \mapsto g^{\mathbf{B}}(\sigma(\vec{y}))]\sigma.$$

Let $t(z, \vec{x}) : U$ be a $(H, z : S)$ -procedure, where \vec{x} is a list of the variables in $Dom(H)$. Let $s = t(g(\vec{y}), \vec{x})$. Then $s : U$ is a H -procedure and

$$\llbracket t : U \rrbracket^{\mathbf{A}} ([z \mapsto g^{\mathbf{A}}(\rho(\vec{y}))]\rho) = \llbracket s : U \rrbracket^{\mathbf{A}} \rho \mathcal{R}_U \llbracket s : U \rrbracket^{\mathbf{B}} \sigma = \llbracket t : U \rrbracket^{\mathbf{B}} ([z \mapsto g^{\mathbf{B}}(\sigma(\vec{y}))]\sigma).$$

Since this holds for every $(H, z : S)$ -procedure t , (4.5) holds by definition of $\mathcal{RE}^\partial(\mathcal{R})$. Hence $\mathcal{RE}^\partial(\mathcal{R})$ has the substitution property.

To verify it has pre-image property, let $H, K \in TCON$ and let h be a context homomorphism from K to H . Let \vec{x} be a list of the variables of K . Let ρ and σ be H -environments in \mathbf{A} and \mathbf{B} , respectively, such that (4.4) holds. We have to show that $\rho \circ h \mathcal{RE}^\partial(\mathcal{R})_K \sigma \circ h$. Let $t(\vec{x}) : T$ be a K -procedure, and let $s = t(h(\vec{x}))$. Then $s : T$ is an H -procedure and

$$\llbracket t : T \rrbracket^{\mathbf{A}} \rho \circ h = \llbracket s : T \rrbracket^{\mathbf{A}} \rho \mathcal{R}_T \llbracket s : T \rrbracket^{\mathbf{B}} \sigma = \llbracket t : T \rrbracket^{\mathbf{B}} \sigma \circ h.$$

This holds for every K -procedure $t : T$. So $\rho \circ h \mathcal{RE}^\partial(\mathcal{R})_K \sigma \circ h$, and $\mathcal{RE}^\partial(\mathcal{R})$ has the pre-image property. \square

An interesting side benefit of this theorem is a completely algebraic characterization of $\mathcal{RE}^\partial(\mathcal{R})$ (involving neither the notions of behavior nor realization) as the largest generalized relation whose projective restriction is included in \mathcal{R} .

Corollary 4.11. *Let $\mathcal{R} \subseteq A \times B$.*

$$\mathcal{RE}^\partial(\mathcal{R}) = \bigcup \{ \mathcal{G} : \mathcal{G} \subseteq ENV^{\mathbf{A} \times \mathbf{B}}, \mathcal{G} \text{ homomorphic, and } \mathcal{G}^- \subseteq \mathcal{R} \}.$$

Proof. Let $\mathcal{H} = \bigcap \{ \mathcal{G} : \mathcal{G} \subseteq ENV^{\mathbf{A} \times \mathbf{B}}, \mathcal{G} \text{ homomorphic, and } \mathcal{G}^- \subseteq \mathcal{R} \}$. The generalized relation $\mathcal{RE}^\partial(\mathcal{R})$ itself is homomorphic by the theorem, and $\mathcal{RE}^\partial(\mathcal{R})^- = \mathcal{BE}(\mathcal{RE}^\partial(\mathcal{R})) \subseteq \mathcal{R}$ by Prop. 4.1. Thus $\mathcal{RE}^\partial(\mathcal{R}) \subseteq \mathcal{H}$. Conversely, suppose \mathcal{G} is a generalized homomorphic relation such that $\mathcal{G}^- \subseteq \mathcal{R}$. Then $\mathcal{G}^- = \mathcal{BE}(\mathcal{G})$ by Prop. 4.1, so $\mathcal{BE}(\mathcal{G}) \subseteq \mathcal{R}$ and hence $\mathcal{G} \subseteq \mathcal{RE}^\partial(\mathcal{R})$ by the basic adjunction. Thus $\mathcal{H} \subseteq \mathcal{RE}^\partial(\mathcal{R})$. \square

The following local analogue of Schoett's theorem (Thm. 4.9) applies to a pair of possibly nonvisible environments.

Theorem 4.12. *Let $H \in TCON$. H -environments ρ and σ over \mathbf{A} and \mathbf{B} are VIS -behaviorally equivalent iff there exists a VIS -identical generalized homomorphic relation \mathcal{G} between \mathbf{A} and \mathbf{B} such that $\rho \mathcal{G} \sigma$.*

Proof. Let ρ and σ be H -environments in \mathbf{A} and \mathbf{B} , respectively. By the local version of the basic adjunction (Cor. 3.10) we have

$$\mathcal{BE}(\rho, \sigma)|_{VIS} \subseteq \mathcal{I} \text{ iff } \rho \mathcal{RE}^\partial(\mathcal{I}^*)_H \sigma.$$

Thus $\mathcal{RE}^\partial(\mathcal{I}^*)$ is the set of all VIS -behaviorally equivalent pairs of environments. $\mathcal{RE}^\partial(\mathcal{I}^*)$ is homomorphic by Thm. 4.10. \square

The following extends the local analogue of Schoett's theorem to algebras. It is the promised exact characterization of VIS -behavioral reducibility and equivalence. The essential idea of the proof is that $\mathcal{RE}^\partial(\mathcal{I}^*)$ is the largest homomorphic generalized relation between \mathbf{A} and \mathbf{B} whose behavior is VIS -identical.

Theorem 4.13. *Let \mathbf{A} and \mathbf{B} be comparable algebras. \mathbf{A} is VIS -behaviorally reducible to \mathbf{B} iff there exists a generalized simulation of \mathbf{B} by \mathbf{A} . The algebras \mathbf{A} and \mathbf{B} are VIS -behaviorally equivalent iff there is a generalized bisimulation between \mathbf{A} and \mathbf{B} .*

Proof. Assume \mathbf{A} is VIS -behaviorally reducible to \mathbf{B} . Then by Def. 3.15(i), for every $H \in TCON$ and $\rho \in ENV_H^{\mathbf{A}}$, there is a $\sigma \in ENV_H^{\mathbf{B}}$ such that $\rho \mathcal{RE}^\partial(\mathcal{I}^*)_H \sigma$. We also have $\mathcal{RE}^\partial(\mathcal{I}^*)$ homomorphic by Thm. 4.10 and VIS -identical by Cor. 3.13(i). Hence by definition $\mathcal{RE}^\partial(\mathcal{I}^*)$ is a generalized simulation of \mathbf{B} by \mathbf{A} .

Suppose that \mathcal{G} is a generalized simulation of \mathbf{B} by \mathbf{A} . Then by definition, \mathcal{G} is homomorphic and VIS -identical. So its behavior is VIS -identical by Cor. 4.2. Thus $\mathcal{G} \subseteq \mathcal{RE}^\partial(\mathcal{I}^*)$ by Cor. 3.13(i). That \mathbf{A} is VIS -behaviorally reducible to \mathbf{B} now follows easily. For suppose $\rho \in ENV_H^{\mathbf{A}}$. Then by the assumption that \mathcal{G} is a generalized simulation, there is a $\sigma \in ENV_H^{\mathbf{B}}$ such that $\rho \mathcal{G}_H \sigma$. Thus $\rho \mathcal{RE}^\partial(\mathcal{I}^*) \sigma$, and hence $\mathcal{BE}(\rho, \sigma)|_{VIS} = \mathcal{I}$ by Cor. 3.13(ii).

The second part of the theorem follows immediately from the first. \square

Term-generated specifications. The previous theorem algebraically characterizes VIS -behavioral equivalence of algebras in terms of generalized homomorphic relations. But it would be preferable to characterize them in terms of standard homomorphic relations. This is not always possible, by Ex. A.1. However, for term-generated algebras we do get such a characterization.

Theorem 4.14. *Assume \mathbf{A} and \mathbf{B} are term-generated. Then \mathbf{A} is VIS -behaviorally reducible to \mathbf{B} iff there exists a standard simulation of \mathbf{A} by \mathbf{B} . Also \mathbf{A} and \mathbf{B} are VIS -behaviorally equivalent iff there is a standard bisimulation between \mathbf{A} and \mathbf{B} .*

Proof Sketch. Assume \mathbf{A} is VIS -behaviorally reducible to \mathbf{B} . We will show that the desired simulation of \mathbf{A} by \mathbf{B} is $\mathcal{BE}(\mathcal{RE}^\partial(\mathcal{I}^*))$. By using properties of \mathcal{I}^* , one can show that $\mathcal{BE}(\mathcal{RE}^\partial(\mathcal{I}^*))$ is homomorphic (this is the sketchy part). By Thm 4.10 and Prop. 4.1 $\mathcal{BE}(\mathcal{RE}^\partial(\mathcal{I}^*)) = \mathcal{RE}^\partial(\mathcal{I}^*)^-$. The behavior $\mathcal{BE}(\mathcal{RE}^\partial(\mathcal{I}^*))$ is VIS -identical by Cor. 4.2. It remains only to verify that $\mathcal{BE}(\mathcal{RE}^\partial(\mathcal{I}^*))$ satisfies condition (i) of Def. 2.13. Let $T \in TYPE$ and $a \in A_T$. Let $x \in VAR$ and $H = x:T$.

Let $\rho_a = \{\langle x, a \rangle\}$. By assumption there is an H -environment $\rho_b = \{\langle x, b \rangle\}$ of \mathbf{B} such that $\rho_a \mathcal{R}\mathcal{E}^\partial(\mathcal{I}^*) \rho_b$. Then $a \mathcal{R}\mathcal{E}^\partial(\mathcal{I}^*)^- b$, and hence $a \mathcal{B}\mathcal{E}(\mathcal{R}\mathcal{E}^\partial(\mathcal{I}^*)) b$. So $\mathcal{B}\mathcal{E}(\mathcal{R}\mathcal{E}^\partial(\mathcal{I}^*))$ is a standard simulation of \mathbf{A} by \mathbf{B} .

Assume now that there exists a standard simulation \mathcal{R} of \mathbf{A} by \mathbf{B} . \mathcal{R} is *VIS*-identical, so $\mathcal{R} \subseteq \mathcal{I}^*$. \mathcal{R} is homomorphic, so \mathcal{R}^+ is homomorphic and $\mathcal{R} = \mathcal{B}\mathcal{E}(\mathcal{R}^+)$ by Thm. 4.3. Thus by the basic adjunction,

$$(4.6) \quad \mathcal{R}^+ \subseteq \mathcal{R}\mathcal{E}^\partial(\mathcal{I}^*).$$

Let $H \in TCON$ and $\rho \in ENV_H^{\mathbf{A}}$. Let $H = \{x_1:T_1, \dots, x_n:T_n\}$ and let $\sigma \in ENV_H^{\mathbf{B}}$ such that, for each $i = 1, \dots, n$, $\rho(x_i) \mathcal{R}_{T_i} \sigma(x_i)$; such a σ exists because \mathcal{R} satisfies condition (i) of Def. 2.13. Thus $\rho \mathcal{R}_H^+ \sigma$ and hence $\rho \mathcal{R}\mathcal{E}^\partial(\mathcal{I}^*) \sigma$ by (4.6). So \mathbf{A} *VIS*-behaviorally reduces to \mathbf{B} .

The second part of the theorem is an immediate consequence of the first part. \square

5. DISCUSSION

In this section we discuss related work, future work, and offer some conclusions.

Related work. In the main body of the paper we have discussed in some detail how our work relates to the work of Schoett [18]. Here we want to make some brief remarks about its connection with the more distantly related, but nevertheless influential, work of Goguen and Meseguer.

The decision to formulate our results as a *behavior-realization adjunction* was inspired by Goguen's [5] categorical theory of automata [5] and its subsequent extension to general modules by Goguen and Meseguer [6, 7]. But the relationship between our theory of behavior, realization, and the associated adjunction and the corresponding theory of Goguen and Meseguer is not straightforward. Roughly speaking, in [6, 7] the behavior of an algebra \mathbf{A} is defined pretty much the way Schoett does, namely, as the abstract function from the set of ground programs to the set of visible data elements that maps each program to its output, when run in \mathbf{A} . (By a ground program we mean a program that has no input variables. There is no loss of generality by restricting to programs of this kind because, since programs can take only visible data as input, and the visible part of \mathbf{A} is term-generated, we can assume the input data is actually part of the program's code.) The realization of an abstract behavior is defined to be any algebra whose concrete behavior coincides with the given abstract one. Let *PROG* stand for the set of all ground programs and *ALG* the class of all Σ -algebras under consideration. The members of *ALG* are assumed to be pairwise comparable in the sense of Def. 1.12. Fix one of them, say \mathbf{B} , so that we can use $\mathbf{B}|_{VIS}$ to represent the visible part of every algebra in *ALG*. The core of the behavior-realization relationship can be thought of as a function of type

$$(5.1) \quad \text{ALG} \times \text{PROG} \rightarrow \mathbf{B}|_{VIS}.$$

Providing *ALG* and $\text{PROG} \rightarrow \mathbf{B}|_{VIS}$ with the structure of categories in a natural way, and then Currying, Goguen and Meseguer get the behavior functor

$$(5.2) \quad E: \text{ALG} \rightarrow (\text{PROG} \rightarrow \mathbf{B}|_{VIS}).$$

This functor turns out to have right adjoint,

$$(5.3) \quad N: (\text{PROC} \rightarrow B|_{\text{VIS}}) \rightarrow \text{ALG},$$

a generalization of the construction of the minimal machine from the theory of finite automata.

The view of behavior and realization in this paper is very different, but it is possible to put it into the same context as that of Goguen and Meseguer's so that we can make some comparisons that might prove useful for finding a common generalization. Let $\text{PROC} = \langle \text{PROC}_H : H \in \text{TCON} \rangle$ where PROC_H is the set of all H -procedures. (Note that PROC is much wider than the class of ground programs because, not only can a procedure return a nonvisible data element, but nonvisible input variables are admitted.) While the algebra \mathbf{A} is treated as a variable in [6, 7], in effect we fix two comparable algebras \mathbf{A} and \mathbf{B} and consider their Cartesian product $\mathbf{A} \times \mathbf{B}$. We have defined a behavior as a special subset of the carrier $A \times B$ of $\mathbf{A} \times \mathbf{B}$, but we could just as well define it as a function of type

$$\text{PROC} \times \mathcal{G} \rightarrow A \times B,$$

where \mathcal{G} is a generalized relation and $\text{PROC} \times \mathcal{G} = \langle \text{PROC}_H \times \mathcal{G}_H : H \in \text{TCON} \rangle$. In order to put our notion of the behavior-dual-realization relationship into a form similar to that of (5.1) a technical transformation is required. In the present context the class of dual realizations can be identified with special subsets of $\text{ENV}^{\mathbf{A} \times \mathbf{B}}$, i.e., members of the powerset $\mathcal{P}(\text{ENV}^{\mathbf{A} \times \mathbf{B}})$. Consider the partial function of type

$$\mathcal{P}(\text{ENV}^{\mathbf{A} \times \mathbf{B}}) \times \text{PROC} \times \text{ENV}^{\mathbf{A} \times \mathbf{B}} \rightarrow A \times B$$

that, for any $\mathcal{G} \in \mathcal{P}(\text{ENV}^{\mathbf{A} \times \mathbf{B}})$, type context H , and $\langle t : T, \langle \rho, \sigma \rangle \rangle \in \text{PROC}_H \times \text{ENV}_H^{\mathbf{A} \times \mathbf{B}}$, takes the value $\langle \llbracket t : T \rrbracket^{\mathbf{A}} \rho, \llbracket t : T \rrbracket^{\mathbf{B}} \sigma \rangle \in A_T \times B_T$ if $\langle \rho, \sigma \rangle \in \mathcal{G}_H$; and is undefined otherwise. Currying once we get a representation of the behavior-dual-realization relationship as a function of type

$$\mathcal{P}(\text{ENV}^{\mathbf{A} \times \mathbf{B}}) \times \text{PROC} \rightarrow (\text{ENV}^{\mathbf{A} \times \mathbf{B}} \rightarrow A \times B),$$

corresponding to (5.1), and Currying once more we get finally the behavior functor

$$\mathcal{BE}: \mathcal{P}(\text{ENV}^{\mathbf{A} \times \mathbf{B}}) \rightarrow (\text{PROC} \rightarrow (\text{ENV}^{\mathbf{A} \times \mathbf{B}} \rightarrow A \times B)),$$

which corresponds to (5.2). This functor also turns out to have a right adjoint

$$(5.4) \quad \mathcal{RE}^\partial: (\text{PROC} \rightarrow (\text{ENV}^{\mathbf{A} \times \mathbf{B}} \rightarrow A \times B)) \rightarrow \mathcal{P}(\text{ENV}^{\mathbf{A} \times \mathbf{B}}).$$

The adjunction of course is just the basic adjunction Cor. 3.9 in another form.

Future work. The main work we plan to do in the future is to use our results to study behavioral subtyping [1, 11]. In earlier work [12] we gave a sufficient algebraic condition for legal behavioral subtyping by using standard homomorphic relations. Using the techniques in this paper, we believe that we can prove a necessary and

sufficient condition for there to be “no surprises” when values of subtypes are used in place of corresponding values of their supertypes. We should also be able to characterize the exact circumstances under which our earlier definition of legal behavioral subtyping is necessary and sufficient.

It should be relatively straightforward to extend the results in this paper to higher-order terms using the appropriate notion of generalized logical relation. Jung and Tiuryn [10] use what appears to be a closely related notion they call “Kripke logical relations of varying arity” to study lambda definability in Henkin models of the simply typed lambda calculus; the idea of such logical relations originated with Sieber [19].

We also plan to consider higher-order terms in the presence of nondeterminism and subtyping, as was done in [12].

Another extension planned is to adapt our results to the study of ADTs with mutable objects (i.e., objects with time-varying state) [3, 13]. Additional questions to investigate are proof-theoretic conditions for behavioral reduction and equivalence, especially for subtyping.

The results of this paper suggest that the general categorical theory of modules presented in [6, 7] might have a useful generalization. It is not clear at this point however what form it should take. As a generalization of the construction of the minimal automata, the functor N of (5.3) gives in some sense the simplest algebra that realizes a given visible behavior. In contrast, if the behavior in (5.4) is specified by fixing the paradigm \mathbf{B} and taking the partial function of type $(\text{PROC} \rightarrow (ENV^{\mathbf{A} \times \mathbf{B}} \rightarrow A \times B))$ to be the one corresponding to the extended visible standard identity relation, then the functor \mathcal{RE}^∂ of (5.4) will give $\mathcal{RE}^\partial(\mathcal{I}^*)$, which may be viewed as the “largest part” of \mathbf{B} that “partially realizes” the given behavior. So dual realization in our sense is *local* in that it can be used to investigate how a particular algebra behaves with respect to any number of given behaviors. This suggests that our theory may be viewed in terms of a comma category formed from the categories ALG and $\text{PROG} \rightarrow A|_{\text{VIS}}$. We hope to explore this possibility in future work.

Conclusions. We have presented a sound and complete model-theoretic technique for proving the correctness of an implementation of a specification. Since we have generalized the notion of observation to allow nonvisible data to be compared, our techniques are broadly applicable. They apply not only to the situation of term-generated and complete specifications, but also to non term-generated and incomplete specifications, such as type parameters.

We have developed the theory of behavior and the notion of dual realization, and studied their properties using their adjunction as our main tool. Behavior specializes to visible behavior, and thus our results include results such as Schoett’s theorem as a special case. Dual realization is a measure of the fragments of algebras that realize a certain behavior, and the dual realization of the generalized identity relation provides the generalized bisimulation that characterizes behavioral equivalence. We developed the theory of generalized relations, because, as we showed, behavioral equivalence cannot be characterized by standard homomorphic relations for incomplete and non-term-generated specifications.

APPENDIX A
INADEQUACY OF STANDARD BISIMULATION FOR
CHARACTERIZING BEHAVIORAL EQUIVALENCE

Schoett's theorem (Thm. 4.9) shows that the existence of a weak standard bisimulation between \mathbf{A} and \mathbf{B} is both necessary and sufficient for weak *VIS*-behavioral equivalence. The following example shows that this fails to hold when the qualifier “weak” is omitted at both places. We then go on to explore in some detail the connection between homomorphic standard and homomorphic generalized relations. Finally, we show by example that not every homomorphic generalized relation is a pointwise extension of a homomorphic standard relation.

Example A.1. The algebras \mathbf{INT} and \mathbf{NAT} , of Examples 1.7 and 1.8, are *VIS*-behaviorally equivalent. However, there is no standard bisimulation between them.

Proof. Recall the generalized homomorphic relation $\mathcal{G}^{\mathbf{P0}}$ of Ex. 2.8. It was shown in Ex. 2.16 that $\mathcal{G}^{\mathbf{P0}}$ is a generalized bisimulation between \mathbf{INT} and \mathbf{NAT} . Hence, these algebras are *VIS*-behaviorally equivalent by Thm. 4.13.

Now suppose, for the sake of contradiction, that \mathcal{R} is a standard bisimulation between \mathbf{INT} and \mathbf{NAT} . Then by definition (2.13), there is a $z \in \mathbf{INT}_{\mathbf{P0}}$, such that $z \mathcal{R}_{\mathbf{P0}} 0$, and there is an $n \in \mathbf{NAT}_{\mathbf{P0}}$ which the integer $z - 1$ simulates: $(z - 1) \mathcal{R}_{\mathbf{P0}} n$. By hypothesis, \mathcal{R} is homomorphic, so

$$\mathit{ff} = \mathit{1eq}^{\mathbf{INT}}(z, z - 1) \mathcal{R}_{\mathbf{Boo1}} \mathit{1eq}^{\mathbf{NAT}}(0, n) = \mathit{tt}.$$

But then \mathcal{R} is not *VIS*-identical, and so cannot be a bisimulation. \square

It might be thought that, even if generalized relations are needed to study *VIS*-behavioral equivalence, perhaps one only needs to use pointwise extensions of standard simulation relations (i.e., relations of the form \mathcal{R}^+). However it is easy to see (with the help of Thm. 4.3) that, if \mathcal{R}^+ is a generalized simulation, then \mathcal{R} must be a standard simulation. So, by the above example, there can be no generalized bisimulation between \mathbf{INT} and \mathbf{NAT} that is the pointwise extension of a standard relation.

In spite of this, there is a close relation between the two notions. Indeed, we show in Thm. A.4 below that every homomorphic generalized relation is the union of the pointwise extensions of homomorphic standard relations. To prove this we need the notion of generated homomorphic generalized relation.

The sets of standard and generalized homomorphic relations between \mathbf{A} and \mathbf{B} are both closed under arbitrary intersection. Moreover, since the total standard relation $A \times B = \langle A_T \times B_T : T \in \mathit{TYPE} \rangle$ and the total generalized relation $ENV^{\mathbf{A} \times \mathbf{B}} = \langle ENV_H^{\mathbf{A}}, ENV_H^{\mathbf{B}} : H \in \mathit{TCON} \rangle$ are both homomorphic, every standard and every generalized relation between \mathbf{A} and \mathbf{B} is included in a smallest homomorphic relation.

Definition A.2 (generated homomorphic relation). Let $\mathcal{G} \subseteq ENV^{\mathbf{A} \times \mathbf{B}}$ be a generalized relation. The *generalized homomorphic relation generated by \mathcal{G}* , denoted $\mathcal{H}(\mathcal{G})$, is defined by:

$$\mathcal{H}(\mathcal{G}) := \bigcap \{ \mathcal{G}' : \mathcal{G} \subseteq \mathcal{G}' \subseteq ENV^{\mathbf{A} \times \mathbf{B}}, \mathcal{G}' \text{ is homomorphic} \}. \quad \square$$

We write $\mathcal{H}(\rho, \sigma)$ as a shorthand for $\mathcal{H}(\{\langle \rho, \sigma \rangle\})$.

Lemma A.3. *Let $K \in TCON$ and $\langle \rho, \sigma \rangle \in ENV_K^{\mathbf{A} \times \mathbf{B}}$ be given. Then*

$$\mathcal{H}(\rho, \sigma) = \mathcal{BE}(\rho, \sigma)^+.$$

Proof. By Prop. 4.5, $\mathcal{BE}(\rho, \sigma)$ is homomorphic. So $\mathcal{BE}(\rho, \sigma)^+$ is homomorphic by Thm. 4.3. Since by definition $\langle \rho, \sigma \rangle \in \mathcal{BE}(\rho, \sigma)^+$, it follows that $\mathcal{H}(\rho, \sigma) \subseteq \mathcal{BE}(\rho, \sigma)^+$.

For the opposite inclusion, suppose for some $L \in TCON$, and $\langle \mu, \nu \rangle \in ENV_L^{\mathbf{A} \times \mathbf{B}}$, $\langle \mu, \nu \rangle \in \mathcal{BE}(\rho, \sigma)^+$. We show that $\langle \mu, \nu \rangle \in \mathcal{H}(\rho, \sigma)$ by induction on the size of $Dom(L)$. To have a stronger inductive hypothesis available for use in the proof, we prove something stronger:

$$(5.5) \quad (\rho \sqcup \mu) \mathcal{H}(\rho, \sigma)_{K \sqcup L} (\sigma \sqcup \nu).$$

Since $\mathcal{H}(\rho, \sigma)$ is a generalized homomorphic relation, if (5.5) holds, then an application of (GHR2) gives the desired inclusion: $\langle \mu, \nu \rangle \in \mathcal{H}(\rho, \sigma)$.

It remains to verify (5.5). For the base case, suppose L is empty. Then $K \sqcup L = K$ and so $\langle \rho \sqcup \mu, \sigma \sqcup \nu \rangle = \langle \rho, \sigma \rangle \in \mathcal{H}(\rho, \sigma)_{K \sqcup L}$. For the inductive case, suppose L is nonempty. Let $x \in Dom(L)$ and denote $K(x)$ by T . Let $L' = L \setminus \{x : T\}$, and let the restrictions of the environments μ and ν to this domain be denoted respectively by μ' and ν' . From the hypothesis $\langle \rho, \sigma \rangle \in \mathcal{BE}(\rho, \sigma)^+$ it follows (by definition of pointwise extension) that $\langle \mu', \nu' \rangle \in \mathcal{BE}(\rho, \sigma)^+$, and so by the induction hypothesis:

$$(5.6) \quad (\rho \sqcup \mu') \mathcal{H}(\rho, \sigma)_{K \sqcup L'} (\sigma \sqcup \nu').$$

Using the hypothesis $\langle \rho, \sigma \rangle \in \mathcal{BE}(\rho, \sigma)^+$ again we get $\langle \mu(x), \nu(x) \rangle \in \mathcal{BE}(\rho, \sigma)$. So there is a K -procedure $t : T$ such that $\llbracket t : T \rrbracket^{\mathbf{A}} \rho = \mu(x)$ and $\llbracket t : T \rrbracket^{\mathbf{B}} \sigma = \nu(x)$. But t is also a $(K \sqcup L')$ -procedure, and so $\llbracket t : T \rrbracket^{\mathbf{A}} (\rho \sqcup \mu') = \mu(x)$ and $\llbracket t : T \rrbracket^{\mathbf{B}} (\sigma \sqcup \nu') = \nu(x)$. Since $\mathcal{H}(\rho, \sigma)$ is homomorphic, and $\rho \sqcup \mu'$ and $\sigma \sqcup \nu'$ are $\mathcal{H}(\rho, \sigma)$ -related by (5.6), it follows from Lem. 2.9 that

$$\begin{aligned} \rho \sqcup \mu &= [x \mapsto (\llbracket t : T \rrbracket^{\mathbf{A}} \rho \sqcup \mu')] (\rho \sqcup \mu') \\ \mathcal{H}(\rho, \sigma)_{K \sqcup L} [x \mapsto (\llbracket t : T \rrbracket^{\mathbf{B}} \sigma \sqcup \nu')] &(\sigma \sqcup \nu') \\ &= \sigma \sqcup \nu. \end{aligned}$$

Thus (5.5) holds, which completes the proof. \square

Theorem A.4. *Let $\mathcal{G} \subseteq ENV^{\mathbf{A} \times \mathbf{B}}$ be given. Then the following are equivalent.*

- (i) \mathcal{G} is homomorphic;
- (ii) $\mathcal{G} = \bigcup_{\langle \rho, \sigma \rangle \in \mathcal{G}} \mathcal{BE}(\rho, \sigma)^+$;
- (iii) $\mathcal{G} = \bigcup \{\mathcal{R}^+ : \mathcal{R} \in X\}$, for some set X of homomorphic standard relations.

Proof. (i) implies (ii): Assume \mathcal{G} is homomorphic. Then for each $\langle \rho, \sigma \rangle \in \mathcal{G}$, $\mathcal{H}(\rho, \sigma) \subseteq \mathcal{G}$. By Lem. A.3, $\mathcal{BE}(\rho, \sigma)^+ \subseteq \mathcal{G}$. So $\bigcup_{\langle \rho, \sigma \rangle \in \mathcal{G}} \mathcal{BE}(\rho, \sigma)^+ \subseteq \mathcal{G}$. The reverse inclusion holds since $\langle \rho, \sigma \rangle \in \mathcal{BE}(\rho, \sigma)^+$ for all $\langle \rho, \sigma \rangle \in \mathcal{G}$.

(ii) implies (iii): By Prop. 4.5, $\mathcal{BE}(\rho, \sigma)$ is a homomorphic standard relation.

(iii) implies (i): Assume that (iii) holds. It follows almost immediately from the definition of generalized homomorphic relations that \mathcal{G} is homomorphic. To

verify (GHR1), suppose that $H \in TCON$ and $\rho \mathcal{G}_H \sigma$. Then $\rho \mathcal{R}_H^+ \sigma$ for some $\mathcal{R} \in X$. Let $g \in OP_n$ and $\vec{T} \rightarrow S$ be a type of g such that $H \vdash \vec{x} : \vec{T}$.

Thus $\rho(\vec{x}) \mathcal{R}_{\vec{T}} \sigma(\vec{x})$. Since \mathcal{R} is homomorphic,

$$g^{\mathbf{A}}(\rho(\vec{x})) \mathcal{R}_S g^{\mathbf{B}}(\sigma(\vec{x})),$$

and hence, for any $y \in VAR \setminus Dom(H)$,

$$[y \mapsto (g^{\mathbf{A}}(\rho(\vec{x})))]\rho \mathcal{R}_{H,y;S}^+ [y \mapsto (g^{\mathbf{B}}(\sigma(\vec{x})))]\sigma.$$

Thus, since $\mathcal{R}^+ \subseteq \mathcal{G}$, we get

$$[y \mapsto (g^{\mathbf{A}}(\rho(\vec{x})))]\rho \mathcal{G}_{H,y;S} [y \mapsto (g^{\mathbf{B}}(\sigma(\vec{x})))]\sigma.$$

Property (GHR2) is established similarly. Thus \mathcal{G} is a generalized homomorphic relation. \square

This theorem does not automatically exclude the possibility that every generalized homomorphic relation is the pointwise extension of a standard relation, but the next example shows this in fact is not the case.

Example A.5. The generalized relation \mathcal{G}^{P0} between **INT** and **NAT** of Ex. 2.8 is a homomorphic generalized relation that is not the pointwise extension of any standard relation.

Proof. \mathcal{G}^{P0} is homomorphic and in fact a generalized bisimulation between **INT** and **NAT** (Exs. 2.8 and 2.16). But it cannot be of the form \mathcal{R}^+ for any standard relation, because, as was observed in the remarks following Ex. A.1, if this were the case, then \mathcal{R} itself would be a standard bisimulation between **INT** and **NAT**, which is impossible by Ex. A.1. \square

REFERENCES

1. P. America, *Designing an Object-Oriented Programming Language with Behavioural Subtyping*, Foundations of Object-Oriented Languages, REX School/Workshop, Noordwijkerhout, The Netherlands, May/June 1990 (J. W. de Bakker and W. P. de Roever and G. Rozenberg, eds.), Lecture Notes in Computer Science, vol. 489, Springer-Verlag, New York, 1991, pp. 60–90.
2. G. Birkhoff, *Lattice theory*, Third Edition, American Mathematical Societ Colloquium Publications, vol. 25, American Mathematical Society, New York, 1967.
3. K. K. Dhara and G. T. Leavens, *Forcing behavioral subtyping through specification inheritance.*, Proceedings 18th International Conference on Software Engineering (Berlin, Germany), IEEE, 1996, pp. 258–267.
4. V. Giarratana, F. Gimona and U. Montanari, *Observability Concepts in Abstract Data Type Specifications*, Mathematical Foundations of Computer Science 1976, Proceedings 5th Symposium, Gdansk (A. Mazurkiewicz, ed.), Lecture Notes in Computer Science, vol. 45, Springer-Verlag, New York, 1976, pp. 576–587.
5. J. A. Goguen, *Realization is Universal*, Math. Systems Theory **6** (1973), 359–374.
6. J. Goguen and J. Meseguer, *Universal Realization, Persistent Interconnection and Implementation of Abstract Modules*, Proceedings, 9th International Conference on Automata, Languages and Programming (M. Nielsen and E. M. Schmidt, eds.), Lecture Notes in Computer Science, vol. 140, Springer-Verlag, New York, 1982, pp. 265–281.

7. J. Goguen and J. Meseguer, *Initiality, Induction and Computability*, Algebraic Methods in Semantics (M. Nivat and J. Reynolds, eds.), Cambridge University Press, Cambridge, 1985, pp. 459–541.
8. J. A. Goguen and J. W. Thatcher and E. G. Wagner, *An Initial Algebra Approach to the Specification, Correctness and Implementation of Abstract Data Types*, Current Trends in Programming Methodology (R. T. Yeh, ed.), vol. 4, Prentice-Hall, Englewood Cliffs, N.J., 1978, pp. 80–149.
9. J. Guttag and J. J. Horning, *The Algebraic Specification of Abstract Data Types*, Acta Informatica **10** (1978), no. 1, 27–52.
10. A. Jung and J. Tiuryn, *A new characterization of lambda definability*, Typed lambda calculus and applications. International Conference on Typed Lambda Calculus and Applications, TLCA '93 March 1993, Utrecht, The Netherlands Proceedings. (M. Bezem and J. F. Groote, eds.), Lecture Notes in Computer Science, vol. 598, Springer-Verlag, Berlin, 1993, pp. 245–257.
11. G. T. Leavens, *Modular Specification and Verification of Object-Oriented Programs*, IEEE Software **8** (July, 1991), no. 4, 72–80.
12. G. T. Leavens and D. Pigozzi, *Typed Homomorphic Relations Extended with Subtypes*, Mathematical Foundations of Programming Semantics '91 (S. Brookes, ed.), Lecture Notes in Computer Science, vol. 598, Springer-Verlag, New York, 1992, pp. 144–167.
13. B. Liskov and J. Wing, *A behavioral notion of subtyping*, TOPLAS, **16** (1994), 1811–1841.
14. S. Mac Lane, *Categories for the Working Mathematician*, Graduate Texts in Mathematics, vol. 5, Springer-Verlag, New York, 1971.
15. T. Nipkow, *Non-deterministic Data Types: Models and Implementations*, Acta Informatica **22** (March, 1986), no. 6, 629–661.
16. J. C. Reynolds, *Using Category Theory to Design Implicit Conversions and Generic Operators*, Semantics-Directed Compiler Generation, Proceedings of a Workshop, Aarhus, Denmark (N. D. Jones, ed.), Lecture Notes in Computer Science, vol. 94, Springer-Verlag, New York, 1980, pp. 211–258.
17. D. Sannella and A. Tarlecki, *On Observational Equivalence and Algebraic Specification*, Journal of Computer and System Sciences **34** (April/June, 1987), no. 2/3, 150–178.
18. O. Schoett, *Behavioural Correctness of Data Representations*, Science of Computer Programming **14** (June, 1990), no. 1, 43–57.
19. K. Sieber, *Reasoning about sequential functions via logical relations*, Applications of Categories in Computer Science. Proceedings of the LMS Symposium, Durham 1991 (M. P. Fourman, P. T. Johnstone, and A. M. Pitts, eds.), London Mathematical Society Lecture Note Series, vol. 177, Cambridge University Press, Cambridge, 1992, pp. 258–269.
20. R. Statman, *Logical Relations and the typed λ -calculus*, Information and Control **65** (May/June, 1985), no. 2/3, 85–97.
21. P. Wadler and S. Blott, *How to make ad-hoc Polymorphism less ad hoc*, Conference Record of the Sixteenth Annual ACM Symposium on Principles of Programming Languages, Austin, Texas, ACM, New York, 1989, pp. 60–76.
22. M. Wand, *Final Algebra Semantics and Data Type Extensions*, Journal of Computer and System Sciences **19** (1979), no. 1, 27–44.
23. M. Wirsing, *Algebraic Specification*, Handbook of Theoretical Computer Science (J. van Leeuwen, ed.), vol. B: Formal Models and Semantics, MIT Press, Cambridge, Mass., 1990, pp. 675–788.
24. M. Wirsing, P. Pepper, H. Partsch, W. Dosch and M. Broy, *On Hierarchies of Abstract Data Types*, Acta Informatica **20** (1983), no. 1, 1–33.

DEPARTMENT OF COMPUTER SCIENCE, IOWA STATE UNIVERSITY, AMES, IOWA 50011-1040
USA

E-mail address: leavens@cs.iastate.edu

DEPARTMENT OF MATHEMATICS, IOWA STATE UNIVERSITY, AMES, IOWA 50011 USA

E-mail address: dpigozzi@iastate.edu