

Program Verification: Lecture 10

José Meseguer

Computer Science Department
University of Illinois at Urbana-Champaign

Unsorted Homomorphisms

Given unsorted Σ -algebras $\mathbb{A} = (A, -_{\mathbb{A}})$ and $\mathbb{B} = (B, -_{\mathbb{B}})$, a Σ -**homomorphism** h from \mathbb{A} to \mathbb{B} , written $h : \mathbb{A} \longrightarrow \mathbb{B}$, is a function $h : A \longrightarrow B$ such that (with s the only sort) we have:

- for each constant $a : nil \longrightarrow s$ in Σ , $h(a_{\mathbb{A}}) = a_{\mathbb{B}}$
(**preservation of constants**)
- for each $f : s \overset{n}{.} s \longrightarrow s$ in Σ and each $(a_1, \dots, a_n) \in A^n$,
we have $h(f_{\mathbb{A}}(a_1, \dots, a_n)) = f_{\mathbb{B}}(h(a_1), \dots, h(a_n))$
(**preservation of operations**)

Example of Unsorted Homomorphism

Ex.10.1. The natural numbers \mathbb{N} , and the natural numbers modulo n , \mathbb{N}_n (for any $n \geq 1$) are all $\Sigma_{\text{NAT-MIXFIX}}$ -algebras (Lecture 3, pages 3–4). Prove in detail that (for any $n \geq 1$) we have a $\Sigma_{\text{NAT-PREFIX}}$ -homomorphism:

$$\text{res}_n : \mathbb{N} \longrightarrow \mathbb{N}_n$$

where res_n sends each number to its residue after dividing by n . For example, $\text{res}_7(23) = 2$, and $\text{res}_5(23) = 3$.

Note that $\Sigma_{\text{NAT-MIXFIX}} = \{0, s, +, *\}$. So you have to **prove** the $\Sigma_{\text{NAT-PREFIX}}$ -homomorphism property of res_n for 0 and for the operations $\{s, +, *\}$.

Examples of Unsorted Homomorphisms (II)

Ex.10.2. Recall (Lecture 3, pgs. 6–8) the powerset algebra $\mathbb{P}(X) = (\mathcal{P}(X), \neg_{\mathbb{P}(X)})$ over the Boolean signature Σ_{BOOL} . Let X and Y be any sets, and let $f : X \longrightarrow Y$ be any function. Prove in detail that the function:

$$f^{-1}[-] : \mathcal{P}(Y) \rightarrow \mathcal{P}(X)$$

defined for any $A \subseteq Y$ by: $f^{-1}[A] = \{x \in X \mid f(x) \in A\}$, is a Σ_{BOOL} -homomorphism $f^{-1}[-] : \mathbb{P}(Y) \rightarrow \mathbb{P}(X)$. Consider also a function $g : Y \longrightarrow Z$. Prove that we have the identity $(f; g)^{-1}[-] = g^{-1}[-]; f^{-1}[-]$, and therefore that $g^{-1}[-]; f^{-1}[-] : \mathcal{P}(Z) \longrightarrow \mathcal{P}(X)$ is also a Σ_{BOOL} -homomorphism from $\mathbb{P}(Z)$ to $\mathbb{P}(X)$.

Many-Sorted Homomorphisms

Given (many-sorted) Σ -algebras $\mathbb{A} = (A, __{\mathbb{A}})$ and $\mathbb{B} = (B, __{\mathbb{B}})$, a Σ -**homomorphism** h from \mathbb{A} to \mathbb{B} , written $h : \mathbb{A} \longrightarrow \mathbb{B}$, is an S -indexed family of functions $h = \{h_s : A_s \longrightarrow B_s\}_{s \in S}$ such that:

- for each constant $a : nil \longrightarrow s$, $h_s(a_{\mathbb{A}}^{nil,s}) = a_{\mathbb{B}}^{nil,s}$
(**preservation of constants**)
- for each $f : w \longrightarrow s$ with $w = s_1 \dots s_n$, $n \geq 1$, and each $(a_1, \dots, a_n) \in A^w$, we have
$$h_s(f_{\mathbb{A}}^{w,s}(a_1, \dots, a_n)) = f_{\mathbb{B}}^{w,s}(h_{s_1}(a_1), \dots, h_{s_n}(a_n))$$
(**preservation of operations**)

Examples of Many-Sorted Homomorphisms

Ex.10.3. Recall the module NAT-LIST in Lecture 2, and the two $\Sigma_{\text{NAT-LIST}}$ -algebras, let us call them \mathbb{A} and \mathbb{B} , defined on pages 4–5 of Lecture 4, namely $\mathbb{A} =$ lists of natural numbers and $\mathbb{B} =$ (finite) sets of natural numbers. Show that there **cannot** be any $\Sigma_{\text{NAT-LIST}}$ -homomorphism $h : \mathbb{A} \longrightarrow \mathbb{B}$.

Ex.10.4. For Σ the signature in picture 4.1, consider the first family of algebras for it described in point 1, pages 5–6 of Lecture 4, namely n -dimensional vector spaces on the rational, the real, or the complex numbers. Let us be specific and fix the reals. Let \mathbb{A} be the 3-dimensional real vector space, and \mathbb{B} the 2-dimensional real vector space. What is then a Σ -homomorphism $h : \mathbb{A} \longrightarrow \mathbb{B}$? Prove that any such homomorphism h can be completely described by

a 2×3 matrix M_h with real coefficients, so that applying to a 3-dimensional vector \vec{v} the homomorphism h , that is, computing $h(\vec{v})$ exactly corresponds to computing the matrix multiplication $\vec{v} \circ M_h$. Generalize this to \mathbb{A} and \mathbb{B} real vector spaces of arbitrary finite dimensions n and m . Generalize it further to rational, resp. complex, vector spaces of any pair of finite dimensions n and m .

Now generalize this even further to characterize by means of matrices **all** Σ -homomorphisms between Σ -algebras in cases 2–3 in page 6 of Lecture 4. Give for each of these cases specific examples of $h : \mathbb{A} \longrightarrow \mathbb{B}$ showing how this works and how h is thus applied to specific elements in the corresponding algebra \mathbb{A} .

Order-Sorted Homomorphisms

For $\Sigma = ((S, <), F)$ an order-sorted signature, and \mathbb{A} and \mathbb{B} order-sorted Σ -algebras, a Σ -homomorphism h from \mathbb{A} to \mathbb{B} , written $h : \mathbb{A} \longrightarrow \mathbb{B}$, is an S -indexed family of functions $h = \{h_s : A_s \longrightarrow B_s\}_{s \in S}$ such that:

- $h : \mathbb{A} \longrightarrow \mathbb{B}$ is a many-sorted (S, F) -homomorphism; and
- if $[s] = [s']$ and $a \in A_s \cap A_{s'}$, then $h_s(a) = h_{s'}(a)$
(agreement on data in the same connected component)

Examples of Order-Sorted Homomorphisms

Ex.10.5. Consider the order-sorted signature Σ of the NAT-LIST-II example in Lecture 2, the two algebras on such a signature, let us call them \mathbb{A} and \mathbb{B} , defined on page 8 of Lecture 4, with \mathbb{A} case (1), and \mathbb{B} case (2). Show that there is **exactly one** order-sorted Σ -homomorphism $h : \mathbb{A} \longrightarrow \mathbb{B}$. Describe such a homomorphism h in complete detail. Show that there **cannot be** any other Σ -homomorphisms $h' : \mathbb{A} \longrightarrow \mathbb{B}$ with $h \neq h'$.

Initiality of the Term Algebra

If a signature is sensible, then different terms denote different things. In the argot of algebraic specifications, this is expressed by saying that the term algebra has **no confusion**.

Furthermore, the term algebra is in some sense **minimal**, since it has only the elements it needs to have to be an algebra: the constants, and the terms needed so that the operations can yield a result; that is why this minimality is expressed saying that it has **no junk**.

Note: In the rest of the course **we will always assume that all signatures are sensible**.

Initiality of the Term Algebra (II)

This minimality means that there is **at most one way** to map homomorphically the elements of \mathbb{T}_Σ to any algebra. And its “no confusion” lack of ambiguity means that such an homomorphic map can **always** be defined.

For example, it couldn't be defined for Σ the non-sensible signature we showed in pg. 16 of Lecture 4 and the Σ -algebra \mathbb{K} with: $K_A = \{a\}$, $K_B = \{b\}$, $K_C = \{c\}$, $K_D = \{d, d'\}$, and with $f^{A,B}(a) = b$, $f^{A,C}(a) = c$, $g^{B,D}(b) = d$, and $g^{C,D}(c) = d'$. Indeed, there is **no** Σ -homomorphism $h : \mathbb{T}_\Sigma \longrightarrow \mathbb{K}$ at all, since $h_D(g(f(a)))$ must be either d or d' . If $h_D(g(f(a))) = d$, then h fails to preserve the operation $g : C \longrightarrow D$, and if $h_D(g(f(a))) = d'$, then h fails to preserve the operation $g : B \longrightarrow D$.

Initiality of the Term Algebra (III)

In summary, the claim is that, if Σ is sensible, then for any Σ -algebra \mathbb{A} there is a **unique** Σ -homomorphism, say, $__{\mathbb{A}} : \mathbb{T}_{\Sigma} \longrightarrow \mathbb{A}$. This is called the **initiality property** of \mathbb{T}_{Σ} . The map $__{\mathbb{A}}$ is the obvious **evaluation function**, mapping each term t to the result of evaluating it in \mathbb{A} . $__{\mathbb{A}}$ is defined inductively in the obvious way:

- for a constant a we define $(a)_{\mathbb{A}} = a_{\mathbb{A}}$, and
- for a term $f(t_1, \dots, t_n)$ we define
$$(f(t_1, \dots, t_n))_{\mathbb{A}} = f_{\mathbb{A}}((t_1)_{\mathbb{A}}, \dots, (t_n)_{\mathbb{A}}).$$

Let us prove it in detail.

Theorem. If Σ is a sensible order-sorted signature, then \mathbb{T}_{Σ} satisfies the initiality property.

Proof of the Initiality Theorem

Proof: For \mathbb{A} any Σ -algebra Let us first prove the uniqueness of $__{\mathbb{A}}$, and then its existence.

Proof of uniqueness. Let us suppose that we have two different homomorphisms $h, h' : \mathbb{T}_{\Sigma} \longrightarrow \mathbb{A}$. We can prove that $h = h'$ by induction on the depth of the terms.

For terms of depth 0 let a be a constant in $T_{\Sigma,s}$. That means that there is a sort $s' \leq s$ with an operator declaration $a : nil \longrightarrow s'$ and therefore, by h and h' being Σ -homomorphisms we must have $h_s(a) = h'_s(a) = a_{\mathbb{A}}^{nil,s'}$.

Proof of the Initiality Theorem (II)

Assume that the equality $h = h'$ holds for terms of depth less or equal to n , and let $f(t_1, \dots, t_n) \in T_{\Sigma, s}$ have depth $n + 1$. That means that there is an operator declaration $f : s_1 \dots s_n \longrightarrow s'$ with $s' \leq s$ and $t_i \in T_{\Sigma, s_i}$, $1 \leq i \leq n$. Again, by h and h' being Σ -homomorphisms we must have:

$$\begin{aligned} h_s(f(t_1, \dots, t_n)) &= \\ &= f_{\mathbb{A}}^{s_1 \dots s_n, s'}(h_{s_1}(t_1), \dots, h_{s_n}(t_n)) \quad (h \text{ homomorphism and } s \leq s') \\ &= f_{\mathbb{A}}^{s_1 \dots s_n, s'}(h'_{s_1}(t_1), \dots, h'_{s_n}(t_n)) \quad (\text{induction hypothesis}) \\ &= h'_s(f(t_1, \dots, t_n)) \quad (h' \text{ homomorphism and } s \leq s'). \end{aligned}$$

Proof of the Initiality Theorem (III)

Proof of Existence. We can both define $-_{\mathbb{A}}$ and show that it is a Σ -homomorphism by induction on the depth of terms. For terms of depth 0, let $a \in T_{\Sigma,s}$ be a constant. That means that there is a sort $s' \leq s$ with an operator declaration $a : nil \longrightarrow s'$; we then define $(a)_{\mathbb{A}_s} = a_{\mathbb{A}}^{nil,s'}$.

Note that the constant a could be subsort-overloaded (cannot be ad-hoc overloaded, since this is ruled out by Σ being sensible) but the above assignment is **well-defined** (does not depend on the particular declaration $a : nil \longrightarrow s'$ chosen), because by our definition of Σ -algebra the interpretations of all subsort overloaded versions of a constant a must coincide in the algebra \mathbb{A} . Furthermore, $-_{\mathbb{A}}$ preserves constants, so it is a Σ -homomorphism.

Proof of the Initiality Theorem (IV)

Assume that $\neg_{\mathbb{A}}$ has already been defined and is a Σ -homomorphism for terms of depth less or equal to n , and let $f(t_1, \dots, t_n) \in T_{\Sigma, s}$ be a term of depth $n + 1$. That means that there is an operator declaration $f : s_1 \dots s_n \longrightarrow s'$ with $s' \leq s$ and $t_i \in T_{\Sigma, s_i}$, $1 \leq i \leq n$. We define

$$(f(t_1, \dots, t_n))_{\mathbb{A}} = f_{\mathbb{A}}^{s_1 \dots s_n, s'}((t_1)_{\mathbb{A}}, \dots, (t_n)_{\mathbb{A}}).$$

Note that, by the induction hypothesis, $\neg_{\mathbb{A}}$ has already been defined for terms of depth less or equal to n and is a Σ -homomorphism on those terms.

Note also that, by the Proof of the Lemma on sensible signatures, for any other $f : s'_1 \dots s'_n \longrightarrow s''$ such that $t_i \in T_{\Sigma, s'_i}$, $1 \leq i \leq n$, we must have, $[s_i] = [s'_i]$, $1 \leq i \leq n$, and $[s'] = [s'']$.

Proof of the Initiality Theorem (V)

Since we have $[s_i] = [s'_i]$, $1 \leq i \leq n$, by definition of order-sorted Σ -homomorphism this then forces,

$$-_{\mathbb{A}_{s_i}}(t_i) = -_{\mathbb{A}_{s'_i}}(t_i), \quad 1 \leq i \leq n.$$

Then, by our definition of Σ -algebra, all subsort overloaded operators must agree on common data, so that we have,

$$f_{\mathbb{A}}^{s_1 \dots s_n, s'}((t_1)_{\mathbb{A}}, \dots, (t_n)_{\mathbb{A}}) = f_{\mathbb{A}}^{s'_1 \dots s'_n, s''}((t_1)_{\mathbb{A}}, \dots, (t_n)_{\mathbb{A}}).$$

Therefore, the definition does not depend on the choice of the subsort overloaded operator. As a consequence, the extension of $-_{\mathbb{A}}$ to the step $n + 1$ is well-defined and, by construction, a Σ -homomorphism, i.e., we have inductively proved the existence of the Σ -homomorphism $-_{\mathbb{A}}$. q.e.d.

The Canonical Term Algebras Revisited

Ex.10.6. Recall the **canonical term algebra**

$\mathbb{C}_{\Sigma/E,B} = (C_{\Sigma/E,B}, \neg_{\mathbb{C}_{\Sigma/E,B}})$, defined in page 17 of Lecture 6 for a functional $\text{fmod } (\Sigma, E \cup B) \text{ endfm}$, where Σ is

B -preregular and satisfies the **Unique Termination**, **Sufficient Completeness** and **Sort Preservation**

requirements.^a By the **Initiality Theorem** we then have a unique Σ -homomorphism $\neg_{\mathbb{C}_{\Sigma/E,B}} : \mathbb{T}_{\Sigma} \longrightarrow \mathbb{C}_{\Sigma/E,B}$. Prove that, as an S -sorted function on S -sorted sets,

$\neg_{\mathbb{C}_{\Sigma/E,B}} : T_{\Sigma} \rightarrow C_{\Sigma/E,B}$ is exactly the S -sorted function:
 $\{T_{\Sigma,s} \ni t \mapsto [t!_{E/B}] \in C_{\Sigma/E,B,s}\}_{s \in S}$ (what Maude's **red** command implements), which we used in defining $\mathbb{C}_{\Sigma/E,B}$.

^aWhich of course can be checked by checking sort-decreasingness, local confluence and termination of \vec{E} modulo B , and sufficient completeness w.r.t. the constructors Ω .

More on Homomorphisms

Ex.10.6. Prove that homomorphisms compose. That is, if $h : \mathbb{A} \longrightarrow \mathbb{B}$ and $g : \mathbb{B} \longrightarrow \mathbb{C}$ are Σ -homomorphisms, then $h;g = \{h_s;g_s\}_{s \in S}$ is a Σ -homomorphism $h;g : \mathbb{A} \longrightarrow \mathbb{C}$.

Ex.10.7. Prove that identities are homomorphisms. That is, given a Σ -algebra $\mathbb{A} = (A, -_{\mathbb{A}})$, the family of identity functions $id_A = \{id_{A_s}\}$ is a Σ -homomorphism $id_A : \mathbb{A} \longrightarrow \mathbb{A}$.

More on Homomorphisms (II)

A Σ -homomorphism $h : \mathbb{A} \longrightarrow \mathbb{B}$ is called an **isomorphism** if there is another Σ -homomorphism $g : \mathbb{B} \longrightarrow \mathbb{A}$ such that $h; g = id_A$ and $g; h = id_B$. We then may use the notation $g = h^{-1}$ and $h = g^{-1}$.

We call a Σ -homomorphism $h : \mathbb{A} \longrightarrow \mathbb{B}$

- **injective** (resp. **surjective**) if for each sort $s \in S$ the function h_s is injective (resp. surjective)
- a **monomorphism** if for any pair of Σ -homomorphisms $g, q : \mathbb{C} \longrightarrow \mathbb{A}$, if $g; h = q; h$ then $g = q$
- an **epimorphism** if for any pair of Σ -homomorphisms $g, q : \mathbb{B} \longrightarrow \mathbb{C}$, if $h; g = h; q$ then $g = q$.

More on Homomorphisms (III)

For example, if \mathbb{N}_{bin} , resp. \mathbb{N}_{dec} , denote the natural numbers with 0, successor, and addition in binary, resp. decimal, representation, we have an obvious binary-to-decimal **isomorphism** $b2d : \mathbb{N}_{bin} \longrightarrow \mathbb{N}_{dec}$ preserving all operations, whose inverse is the decimal-to-binary **isomorphism**, $d2b : \mathbb{N}_{bin} \longrightarrow \mathbb{N}_{dec}$. Of course, $d2b; b2d = id_{\mathbb{N}_{dec}}$, and $b2d; d2b = id_{\mathbb{N}_{bin}}$.

For \mathbb{N}_n the residue classes modulo n , the reminder function $\mathbb{N} \xrightarrow{rem_n} \mathbb{N}_n$ is a **surjective** homomorphism for Σ containing, say, 0, 1, +, \times .

Similarly, for \mathbb{Z}_{dec} the integers in decimal notation, the inclusion $j : \mathbb{N}_{dec} \hookrightarrow \mathbb{Z}_{dec}$ is an **injective** homomorphism preserving all shared operations: 0, 1, +, \times , etc.

Theorem: All Initial Algebras Are Isomorphic

Proof: Suppose \mathbb{A} and \mathbb{B} are Σ -algebras and both satisfy the initiality property of having a unique Σ -homomorphism to any other algebra. In particular, we have unique homomorphisms,

$$h : \mathbb{A} \longrightarrow \mathbb{B} \quad g : \mathbb{B} \longrightarrow \mathbb{A}$$

and therefore a composed homomorphism

$$h;g : \mathbb{A} \longrightarrow \mathbb{B} \longrightarrow \mathbb{A}$$

but we also have the identity homomorphism id_A , which by uniqueness forces $h;g = id_A$. Interchanging the role of \mathbb{A} and \mathbb{B} we also get, $g;h = id_B$. q.e.d.

Evaluating Program Expressions

Q1: Can we **model** the **evaluation of expressions** in a programming language using **initial algebras**?

A1: We first of all need a **signature** Σ of operations.

For example, Σ could be a signature for integer operations, and/or Boolean operations, and/or real number operations (typically using a floating point representation).

Assume a programming language in which we only have integers and integer operations (note that we can encode true and false as, respectively, 0 and 1). In this case Σ can be unsorted and have two constants, 0 and 1, and three binary function symbols: $_ + _$, $_ - _$, and $_ * _$.

Evaluating Program Expressions (II)

Q2: What else do we need?

A2: We need a set X of **variables** appearing on our expressions. This means that we need to extend Σ to $\Sigma(X)$, so that our program **expressions** will be **terms** $t \in T_{\Sigma(X)}$.

Q3: And what else do we need if we want to **evaluate** such expressions?

A3: We of course need a Σ -**algebra** in which they will be evaluated. For integers expressions this is the algebra $\mathbb{Z} = (\mathbb{Z}, -_{\mathbb{Z}})$ of the integers with $+, *, -, 0, 1$.

Evaluating Program Expressions (III)

Q4: And what else do we need?

A4: Since expression evaluation **depends** on the memory state, we need to **model mathematically** memory states.

Q5: And how can we model **memory states**?

A5: Assuming programs with just global variables, a memory state for arithmetic expressions is just a **function** $m : X \rightarrow \mathbb{Z}$. This is a special instance of the general notions of an **assignment** of values to variables in an **algebra**.

Assignments

Given variables in $X = \{X_s\}$ we will often be interested in **assignments** (also called **valuations**) of data elements in a given Σ -algebra $\mathbb{A} = (A, -_{\mathbb{A}})$ to those variables. Of course, if $x \in X_s$ then the value, say $a(x)$, assigned to x should be an element of A_s . That is the assignments should be **well-sorted**. This can be made precise by defining an **assignment** to the variables X in a Σ -algebra $\mathbb{A} = (A, -_{\mathbb{A}})$ to be an S -indexed family of functions, $a = \{a_s : X_s \longrightarrow A_s\}_{s \in S}$, denoted $a : X \longrightarrow A$.

Often what we want to do with such assignments is to **extend** them from variables to terms on such variables in the obvious, homomorphic way.

Evaluating Program Expressions (VI)

Q6: Now that we have everything we need, how can **evaluation of arithmetic expressions** be precisely defined relative to a memory (state) $m : X \rightarrow \mathbb{Z}$?

A6: As a function $_{(\mathbb{Z}, m)} : T_{\Sigma(X)} \rightarrow \mathbb{Z}$ defined inductively by:

1. $x_{(\mathbb{Z}, m)} = m(x)$ for $x \in X$
2. $0_{(\mathbb{Z}, m)} = 0 \in \mathbb{Z}$, $1_{(\mathbb{Z}, m)} = 1 \in \mathbb{Z}$
3. $f(t, t')_{(\mathbb{Z}, m)} = f_{\mathbb{Z}}(t_{(\mathbb{Z}, m)}, t'_{(\mathbb{Z}, m)})$ for $f \in \{+, *, -\}$.

Evaluating Program Expressions (VII)

Q7: Conditions (2)–(3) show that $_{-(\mathbb{Z}, m)}$ is a Σ -homomorphism. What about condition (1)?

A7: Condition (1) plus (2)–(3) show that it is a $\Sigma(X)$ -homomorphism, when we **extend** the algebra \mathbb{Z} of the integers with the **additional constants** X , where each $x \in X$ is interpreted in \mathbb{Z} as $m(x)$. Denote this extended $\Sigma(X)$ -algebra (\mathbb{Z}, m) . Then the evaluation of arithmetic expressions is the **unique** $\Sigma(X)$ -**homomorphism**:

$$_{-(\mathbb{Z}, m)} : \mathbb{T}_{\Sigma(X)} \rightarrow (\mathbb{Z}, m)$$

to the $\Sigma(X)$ -algebra (\mathbb{Z}, m) (extending the Σ -algebra \mathbb{Z} with memory m) ensured by the initiality of $\mathbb{T}_{\Sigma(X)}$.

Exercises

Ex.10.8. Show that a homomorphism is injective iff it is a monomorphism. Prove that every surjective homomorphism is an epimorphism. Construct an epimorphism that is not surjective.

Ex.10.9. Show that any many-sorted Σ -homomorphism that is surjective and injective is an isomorphism.

Construct an order-sorted homomorphism that is surjective and injective but is not an isomorphism. Give a sufficient condition on the poset (S, \leq) (more general of course than being a discrete poset, since that is the many-sorted case) so that h is an isomorphism iff h is surjective and injective.

Exercises (II)

Ex.10.10. Prove that if an algebra \mathbb{J} is isomorphic to an initial algebra \mathbb{I} , then \mathbb{J} itself is initial.

Ex.10.11. Show that the natural numbers in Peano notation (zero and successor) and in base 2 are isomorphic Σ -algebras (both initial) for Σ the signature with one sort `Natural` and zero and successor operations.