Algebra Univers. (2020) 81:18
© 2020 The Author(s)
1420-8911/20/020001-14
published online March 30, 2020
https://doi.org/10.1007/s00012-020-00654-w

Algebra Universalis

Connected monads weakly preserve products

H. Peter Gumm

Dedicated to Ralph Freese, Bill Lampe, and J.B. Nation.

Abstract. If F is a (not necessarily associative) monad on *Set*, then the natural transformation $F(A\times B) \to F(A)\times F(B)$ is surjective if and only if $F(\mathbf{1}) = \mathbf{1}$. Specializing F to $F_{\mathcal{V}}$, the free algebra functor for a variety $\mathcal{V}$, this result generalizes and clarifies an observation by Dent, Kearnes and Szendrei in 2012.

Mathematics Subject Classification. 18A22, 08B10, 08B20.

Keywords. Idempotent free algebra, Connected monad, Preservation of products.

1. Introduction

A key observation in [2] by T. Dent, K. Kearnes, and Á. Szendrei is that for any variety $\mathcal{V}$ with idempotent operations each set theoretic product decomposition

$$d : \{x, y, z, u\} \twoheadrightarrow \{a, b\} \times \{a, b\}$$

extends to a surjective homomorphism

$$\delta : F_{\mathcal{V}}(\{x, y, z, u\}) \twoheadrightarrow F_{\mathcal{V}}(\{a, b\}) \times F_{\mathcal{V}}(\{a, b\}) \tag{1.1}$$

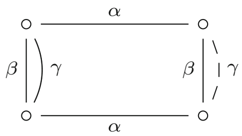
from the 4-generated free algebra in $\mathcal{V}$ to the square of the 2-generated one.

This fact has an interesting geometric interpretation, which is relevant in the study of congruence modularity. The shifting lemma from [9], which is concerned with shifting a congruence γ from one side of an $\alpha-\beta$-parallelogram to the opposite side modulo $\alpha\wedge\beta$, can be specialized to axis-parallel rectangles

Presented by W. DeMeo.
This article is part of the topical collection "Algebras and Lattices in Hawaii" edited by W. DeMeo.

Birkhäuser

inside a product of algebras where α and β are in fact kernels of the projections and γ a factor congruence.



Surjectivity of the above map implies that the projections on the image commute, and since $\operatorname{Ker}\delta = \alpha \wedge \beta$, it follows that α and β also commute in the preimage. In particular, therefore, the shifting lemma, which in [9] is the major geometrical tool for studying congruence modularity, is only needed in situations of permuting congruence relations α and β. The restriction to idempotent varieties in these studies is not severe, since a variety is congruence modular iff its idempotent reduct is congruence modular.

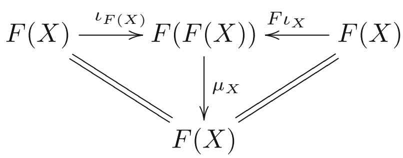
Variations of the shifting lemma (e.g. [1]) and, more recently, categorical generalizations as in [3] suggest to investigate the situation in a more general context. In this note, therefore, rather than exploring further ramifications of the above observation, we explore the abstract reasons behind the surjectivity of δ in (1.1). It turns out that we can deal with this in a framework which is more abstract than universal algebras and varieties. We are rather considering (not necessarily associative) *Set*-monads F, of which the functor $F_{\mathcal{V}}$, which associates with a set X the free algebra $F_{\mathcal{V}}(X)$ and with a map $g : X \to Y$ its unique homomorphic extension $\bar{g} : F_{\mathcal{V}}(X) \to F_{\mathcal{V}}(Y)$, is just an example.

2. Monads and main result

Monads on a category $\mathcal{C}$ are functors $F : \mathcal{C} \to \mathcal{C}$ together with two natural transformations $\iota : Id \to F$ and $\mu : F \circ F \to F$, satisfying two unit laws and an associative law. Our results will even hold for nonassociative monads, so skipping the associative law, we shall only state the unit laws:

$$\mu_X \circ \iota_{F(X)} = id_{F(X)} = \mu_X \circ F\iota_X \tag{2.1}$$

Equations (2.1) are usually expressed as a commutative diagram:



Rather easy examples of monads on the category of sets are obtained from collection data types in programming, such as $List\langle X\rangle$, $Set\langle X\rangle$ or $Tree\langle X\rangle$; see also [11]. In popular programming languages, $List\langle X\rangle$ denotes the type of lists of elements from a base type X. Given a function $g : X \to Y$, the function $map(g) : List\langle X\rangle \to List\langle Y\rangle$ which sends $[x_1, \ldots, x_n] \in List\langle X\rangle$ to the list $[g(x_1), \ldots, g(x_n)] \in List\langle Y\rangle$ represents the action of the functor *List* on maps. In mathematical notation we write $(List\ g)$ rather than $map(g)$. Obviously,

$map(f \circ g) = map(f) \circ map(g)$ and $map(id_X) = id_{List\langle X\rangle}$, so the pair $List\langle -\rangle$ with map indeed establishes a functor.

For $List$ to be a monad, we need a natural transformation $\iota : Id \to List$, as well as a "multiplication" $\mu : List \circ List \to List$. The former can be chosen as the *singleton* operator with $\iota_X : X \to List\langle X\rangle$ sending any $x \in X$ to the one-element list $[x]$.

The monad multiplication μ is for each type X defined as

$$\mu_X : List\langle List\langle X\rangle\rangle \to List\langle X\rangle,$$

taking a list of lists $[l_1, \ldots, l_n]$ and appending them into a single list $l_1 + \cdots + l_n$. Programmers call this operation "*flatten*". The unit laws then state that for each list $l = [x_1, \ldots, x_n] \in List\langle X\rangle$ we should have

$$flatten([\,[x_1, \ldots, x_n]\,]) = [x_1, \ldots, x_n] = flatten([\,[x_1], \ldots, [x_n]\,]),$$

which is obvious. Not all monads arise from collection classes, and other uses of monads have all but revolutionized functional programming, see e.g. [12] or [15].

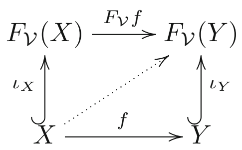
Relevant for universal algebraists is the fact that for every variety $\mathcal{V}$ the construction of the free algebra $F_{\mathcal{V}}(X)$ over a set X is a monadic functor. In this case, $\iota_X : X \to F_{\mathcal{V}}(X)$ is the inclusion of variables, or rather their interpretations as $\mathcal{V}-$terms.

The defining property of $F_{\mathcal{V}}(X)$ states that each map $g : X \to A$ for $A \in \mathcal{V}$ has a unique homomorphic extension $\bar{g} : F_{\mathcal{V}}(X) \to A$.

From a map $f : X \to Y$, we therefore obtain the homomorphism

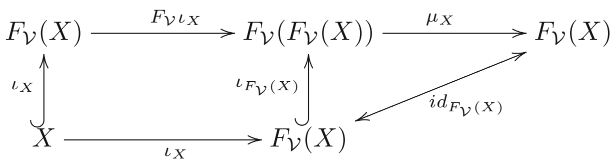
$$F_{\mathcal{V}} f : F_{\mathcal{V}}(X) \to F_{\mathcal{V}}(Y)$$

as the unique homomorphic extension of the composition $\iota_Y \circ f : X \to F_{\mathcal{V}}(Y)$:



The flattening map $\mu : F_{\mathcal{V}}(F_{\mathcal{V}}(X)) \to F_{\mathcal{V}}(X)$ can be considered as term composition: a term $t(t_1, \ldots, t_n)$, whose argument positions have been filled by other terms, is interpreted as an honest $\mathcal{V}$-term. To make this precise, consider the diagram below, in which $F_{\mathcal{V}}(X)$ appears in two roles – in the top row as an algebra and in the bottom row as a set of free variables for $F_{\mathcal{V}}(F_{\mathcal{V}}(X))$.

The left square is obtained by instantiating the previous diagram with $Y := F_{\mathcal{V}}(X)$ and $f := \iota_X$. The right hand triangle defines μ_X as the homomorphic extension of the identity map $id_{F_{\mathcal{V}}(X)}$ from $F_{\mathcal{V}}(X)$, considered as set of free variables for $F_{\mathcal{V}}(F_{\mathcal{V}}(X))$, to $F_{\mathcal{V}}(X)$ considered as a $\mathcal{V}$-algebra.



The first monad equation immediately follows from the definition of μ, and the second equation

$$\mu_X \circ (F_{\mathcal{V}} \iota_X) = id_{F_{\mathcal{V}}(X)}$$

follows from the fact that both the left hand side and the right hand side of this equation are homomorphic extensions of $\iota_X : X \to F_{\mathcal{V}}(X)$, as can be read from the diagram, so they must be equal.

The earlier mentioned examples $Tree\langle X\rangle$, $List\langle X\rangle$, and $Set\langle X\rangle$, just correspond to the free groupoid, the free semigroup, and the free semilattice over the set X of generators, and are themselves instances of this scheme.

We are now ready to state our main result.

Theorem 2.1. A (not necessarily associative) Set-monad F weakly preserves products if and only if $F(\mathbf{1}) \cong \mathbf{1}$.

It will be easy to see (Lemma 4.6 below) that F *weakly preserves the product* $A_1 \times A_2$ if and only if the canonical morphism $\delta = (F\pi_1, F\pi_2)$ in the below diagram is epi:

$$\begin{array}{ccc} F(A_1 \times A_2) & \xrightarrow{\quad\delta\quad} & F(A_1) \times F(A_2) \\ & {\scriptstyle F\pi_i} \searrow \quad \swarrow {\scriptstyle \eta_i} & \\ & F(A_i) & \end{array} \tag{2.2}$$

The starting point of our discussion, (1.1) from [2], is therefore seen to represent an instance of this result when setting $A_1 = A_2 = \{a, b\}$ and $F = F_{\mathcal{V}}$. But before coming to its proof we need a few preparations.

3. Connected functors

Put $\mathbf{1} = \{0\}$ and for any set X denote by $!_X$ the unique (terminal) map from X to $\mathbf{1}$. A *Set*-functor F is called *connected* if $F(\mathbf{1}) \cong \mathbf{1}$. Given a variety $\mathcal{V}$, the functor $F_{\mathcal{V}}$ is connected if and only if $\mathcal{V}$ is idempotent.

It is well known, see [14], that every *Set*-Functor F can be constructed as a sum of connected functors:

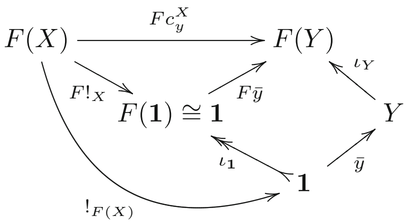
$$F = \Sigma_{e \in F(\mathbf{1})} F_e.$$

For $e \in F(\mathbf{1})$ one simply puts $F_e(X) = \{u \in F(X) \mid (F!_X)(u) = e\}$. On maps $f : X \to Y$, each subfunctor F_e is just the domain-codomain-restriction of Ff to $F_e(X)$.

In the following we denote by $c_y^X : X \to Y$ or, if X is clear, simply by c_y, the constant map with value $y \in Y$. We shall need the following lemma:

Lemma 3.1. If F is a connected functor, then Fc_y^X is a constant map. Whenever $\iota : Id \to F$ is a natural transformation, then $Fc_y^X = c_{\iota_Y(y)}^{F(X)}$.

Proof. For $y \in Y$, denote by $\bar{y} : \mathbf{1} \to Y$ the constant map with value y. Observe, that an arbitrary map f is constant if and only if it factors through $\mathbf{1}$, i.e. $c_y^X = \bar{y} \circ !_X$. Applying F and adding the natural transformation ι into the picture,



we obtain:

$$\begin{aligned} Fc_y^X &= F\bar{y} \circ F!_X \\ &= F\bar{y} \circ \iota_{\mathbf{1}} \circ !_{F(X)} \\ &= \iota_Y \circ \bar{y} \circ !_{F(X)} \\ &= \overline{\iota_Y(y)} \circ !_{F(X)} \\ &= c_{\iota_Y(y)}^{F(X)}. \end{aligned}$$

□

In the above, we have seen that connected functors preserve constant maps. It might be interesting to remark that this very property characterizes connected functors:

Corollary 3.2. *A functor F is connected if and only if for every constant morphism c_y the morphism Fc_y is constant, again.*

Proof. Suppose that F preserves constant maps. As $id_{\mathbf{1}}$ is constant, $F(id_{\mathbf{1}}) = id_{F(\mathbf{1})}$ must also be constant, which implies $F(\mathbf{1}) \cong \mathbf{1}$. □

In general, the elements of $F(\mathbf{1})$ correspond uniquely to the natural transformations between the identity functor Id and F. This can be seen by instantiating the Yoneda Lemma

$$nat(Hom(A, -), F) \cong F(A) \tag{3.1}$$

with $A = \mathbf{1}$. Therefore we note:

Corollary 3.3. *A monad (F, ι, μ) is connected if and only if ι is the only transformation from the identity functor to F.*

Definition 3.4. Let $\mathcal{C}_{\mathbf{1}}$ be the constant functor with $\mathcal{C}_{\mathbf{1}}(X) = \mathbf{1}$ for all X and $\mathcal{C}_{\mathbf{1}} f = id_{\mathbf{1}}$ for all f. We say that a functor F *possesses a constant*, if there is a transformation from $\mathcal{C}_{\mathbf{1}}$ to F which is natural, except perhaps at $X = \emptyset$.

Clearly, each element of $F(\emptyset)$ gives rise to a constant, but not conversely, since there is nothing to stop us from changing F only on the empty set $\emptyset$ and on empty maps $\emptyset_X : \emptyset \to X$ by choosing any $U \subseteq F(\emptyset)$ and redefining $F'(\emptyset) := U$ as well as $F'\emptyset_X = F\emptyset_X \circ \subseteq_U^X$. For that reason we do not require naturality at $\emptyset$ in the above definition.

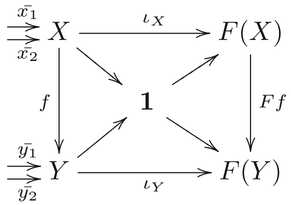
We shall need the following observation:

Lemma 3.5. A connected functor either possesses a constant or it has the identity functor as a subfunctor.

Proof. By the Yoneda Lemma, there is exactly one natural transformation $\iota : Id \to F$. Assume that some ι_X is not injective, then there are $x_1 \neq x_2 \in X$ with $\iota_X(x_1) = \iota_X(x_2)$. Given an arbitrary Y with $y_1, y_2 \in Y$, consider a map $f : X \to Y$with $f(x_1) = y_1$ and $f(x_2) = y_2$. By naturality,

$$\iota_Y(y_1) = \iota_Y(f(x_1)) = Ff \circ \iota_X(x_1) = Ff \circ \iota_X(x_2) = \iota_Y(y_2),$$

hence each ι_Y is constant and therefore factors through $\mathbf{1}$. This makes the upper and lower triangle inside the following naturality square commute, too.



The left triangle commutes since $\mathbf{1}$ is terminal. If $X \neq \emptyset$, the terminal map $!_X : X \to \mathbf{1}$ is epi, from which we now conclude that the right triangle commutes as well, except, possibly, when $X = \emptyset$. Thus F possesses a constant. □

4. Preservation properties

We are concerned with the question, under which conditions the δ in equation (1.1) is epi. Therefore, we take a look at the canonical map $\delta = (F\pi_1, F\pi_2) : F(A_1 \times A_2) \to F(A_1) \times F(A_2)$ which arises from the commutative diagram (2.2), where π_i, resp η_i, denote the canonical component projections.

The first thing to observe is:

Lemma 4.1. $\delta = (F\pi_1, F\pi_2) : F(A_1 \times A_2) \to FA_1 \times FA_2$ is natural in each component.

Proof. Assume $f : A_1 \to A_1'$ and $g : A_2 \to A_2'$ are given. We want to show that the following diagram commutes:

$$\begin{array}{ccc} F(A_1 \times A_2) & \xrightarrow{(F\pi_1, F\pi_2)} & F(A_1) \times F(A_2) \\ {\scriptstyle F(f\times g)}\downarrow & & \downarrow{\scriptstyle Ff \times Fg} \\ F(A_1' \times A_2') & \xrightarrow{(F\pi_1', F\pi_2')} & F(A_1') \times F(A_2') \end{array}$$

We calculate:

$$\begin{aligned}((Ff \times Fg) \circ (F\pi_1, F\pi_2))(u) &= (Ff \times Fg)((F\pi_1)(u), (F\pi_2)(u)) \\ &= ((Ff \circ F\pi_1)(u), (Fg \circ F\pi_2)(u)) \\ &= (F(f \circ \pi_1)(u), F(g \circ \pi_2)(u)) \\ &= (F(\pi_1' \circ f \times g)(u), F(\pi_2' \circ f \times g)(u)) \\ &= ((F\pi_1' \circ F(f \times g))(u), (F\pi_2' \circ F(f \times g))(u)) \\ &= (F(\pi_1')(F(f \times g)(u)), F(\pi_2')(F(f \times g)(u))) \\ &= (F\pi_1', F\pi_2')(F(f \times g)(u)) \\ &= ((F\pi_1', F\pi_2') \circ F(f \times g))(u). \end{aligned}$$ □

Notice that in order for δ to be surjective, the functor F must be connected or trivial.

Lemma 4.2. If the canonical decomposition as in Theorem 2.1 is always epi, then either $F(\mathbf{1}) \cong \mathbf{1}$ or F is the trivial functor with constant value $\emptyset$.

Proof. For the projections $\pi_1, \pi_2 : \mathbf{1} \times \mathbf{1} \to \mathbf{1}$ we have $\pi_1 = \pi_2$, since $\mathbf{1}$ is a terminal object, hence also $F\pi_1 = F\pi_2$. Let η_1, η_2 be the projections from the product $F(\mathbf{1}) \times F(\mathbf{1})$ to its components. Then

$$\eta_1 \circ (F\pi_1, F\pi_2) = F\pi_1 = F\pi_2 = \eta_2 \circ (F\pi_1, F\pi_2).$$

By assumption, $\delta = (F\pi_1, F\pi_2)$ is epi, so $\eta_1 = \eta_2$. For arbitrary $a, b \in F(\mathbf{1})$ then $(a, b) \in F(\mathbf{1}) \times F(\mathbf{1})$, so

$$a = \eta_1(a, b) = \eta_2(a, b) = b.$$

So $F(\mathbf{1})$ either has just one element, or $F(\mathbf{1}) = \emptyset$. In the latter case, for each set X the map $!_X : X \to \mathbf{1}$ should yield a map $F!_X : F(X) \to F(\mathbf{1})$, so $F(\mathbf{1}) = \emptyset$ implies $F(X) = \emptyset$. □

Next, recall some elementary categorical notions.

Definition 4.3. Given objects A_1, A_2 in a category $\mathcal{C}$, a *product of* A_1 *and* A_2 is an object P together with morphisms $p_i : P \to A_i$, such that for any "competitor", i.e. for any object Q with morphisms $q_i : Q \to A_i$, there exists *a unique* morphism $d : Q \to P$, such that $q_i = p_i \circ d$ for $i = 1, 2$. Products, if they exist, are unique up to isomorphism and are commonly written $A_1 \times A_2$.

Similarly, given morphisms $f_1 : A_1 \to B$ and $f_2 : A_2 \to B$ with common codomain B, their *pullback* is defined to be a pair of maps $p_1 : P \to A_1$ and $p_2 : P \to A_2$ with common domain P such that

$$f_1 \circ p_1 = f_2 \circ p_2$$

and for each "competitor", i.e. each object Q with morphisms $q_1 : Q \to A_1$ and $q_2 : Q \to A_2$ also satisfying $f_1 \circ q_1 = f_2 \circ q_2$ there exists *a unique* morphism $d : Q \to P$ so that $p_i \circ d = q_i$ for $i = 1, 2$ (see Figure 1).

In both definitions, if we drop the uniqueness requirement, we obtain the definition of *weak product*, resp. *weak pullback*.

FIGURE 1. (Weak) pullback and (weak) product

Notice that in case there exists a terminal object **1**, the product of A_1 with A_2 is the same as the pullback of the terminal morphisms $!_{A_i} : A_i \to \mathbf{1}$.

Weak products (weak pullbacks) arise from right invertible morphisms into products (pullbacks):

Lemma 4.4. If (P, p_1, p_2) is a product (resp. pullback), then (W, w_1, w_2) is a weak product (resp. weak pullback) if and only if there is a right invertible $w : W \to P$ such that $w_i = p_i \circ w$.

Proof. If w has a right inverse e, and (Q, q_1, q_2) is a competitor to W, then it is also a competitor to P, hence there is a morphism $d : Q \to P$ with $q_i = p_i \circ d$. Then $e \circ d$ is the required morphism to W. Indeed,

$$w_i \circ (e \circ d) = p_i \circ w \circ e \circ d = p_i \circ d = q_i.$$

Conversely, assume that (W, w_1, w_2) is a weak product, then both W and P are competitors to each other, yielding both a morphism $w : W \to P$ with $w_i = p_i \circ w$ and a morphism $e : P \to W$ with $p_i = w_i \circ e$.

Now (P, p_1, p_2) is also a competitor to itself, yet both $p_i \circ (w \circ e) = p_i$ and $p_i \circ id_P = p_i$ for $i = 1, 2$. By uniqueness it follows that $w \circ e = id_P$, so w is indeed right invertible. (The same proof works for the case of weak pullbacks). □

Definition 4.5. Let $F : \mathcal{C} \to \mathcal{D}$ be a functor. We say that F *weakly preserves* products (pullbacks) if whenever (P, p_1, p_2) is a product (pullback), then its image $(F(P), Fp_1, Fp_2)$ is a weak product (weak pullback).

It is well known that a functor weakly preserves a limit L if and only it preserves weak limits, see e.g. [5]. Surjective maps are right invertible, so regarding (1.1) or its more general formulation (2.2), we now arrive at the following relevant observation:

Lemma 4.6. The canonical map δ in (2.2) is epi if and only if F weakly preserves the product $(A_1 \times A_2, \pi_1, \pi_2)$.

Whereas the above mentioned result of [2], in which the monad F is the free-algebra-functor $F_{\mathcal{V}}$, served a purely universal algebraic purpose, it also has an interesting coalgebraic interpretation. It is well known that coalgebraic properties of classes of F-coalgebras are to a large degree determined by *weak pullback preservation* properties of the functor F, which serves as a *type* or *signature* for a class $Coalg_F$ of coalgebras. Prominent structure theoretic properties can be derived from the assumptions that F weakly preserves pullbacks of preimages, kernel pairs or both, see e.g. [4,5,6,7,8,13]. Here we add one more property to this list: preservation of pullbacks of constant maps.

Theorem 4.7. Let F be a nontrivial functor. Then the following are equivalent:

(1) F has no constant and weakly preserves products.
(2) F is connected and weakly preserves pullbacks of constant maps.

Proof. If F is nontrivial and weakly preserves the product $\mathbf{1} \times \mathbf{1} \cong \mathbf{1}$, then F is connected as a consequence of Lemma 4.2. Since F has no constants, $F(\emptyset) = \emptyset$ and moreover Lemma 3.5 provides Id as a subfunctor of F. Thus we obtain a natural transformation $\iota : Id \to F$ which is injective in each component.

Let now $c_{y_i}^{X_i} : X_i \to Y$ for $i = 1, 2$ be constant maps with $y_i \in Y$. Applying F, Lemma 3.1 yields $Fc_{y_i}^{X_i} = c_{\iota_Y(y_i)}^{F(X_i)}$ for $i = 1, 2$.

If $y_1 = y_2$ then the pullback of the $c_{y_i}^{X_i}$ is simply $(X_1 \times X_2, \pi_1, \pi_2)$. The $Fc_{y_i}^{X_i}$ are constant maps with the same target value $\iota_Y(y_1) = \iota_Y(y_2)$, so their pullback is the product $F(X_1) \times F(X_2)$ with canonical projections $\eta_i : F(X_1) \times F(X_2) \to F(X_i)$. By assumption, F weakly preserves products, which gives us a surjective canonical map $\delta : F(X_1 \times X_2) \to F(X_1) \times F(X_2)$ with $F\pi_i = \eta_i \circ \delta$, so Lemma 4.4 ensures that $(F(X_1 \times X_2), F\pi_1, F\pi_2)$ is a weak pullback of the $Fc_{y_i}^{X_i}$.

If $y_1 \neq y_2$, then the pullback of the $c_{y_i}^{X_i}$ is $(\emptyset, \emptyset_{X_1}, \emptyset_{X_2})$, the empty set $\emptyset$ with empty mappings $\emptyset_{X_i} : \emptyset \to X_i$. Since ι_Y is injective, the Fc_{y_1} are constant maps with disjoint images, too, consequently their pullback is $(\emptyset, \emptyset_{F(X_1)}, \emptyset_{F(X_2)})$. This is the same we would obtain by applying F to the pullback of the c_{y_i}, taking into account that $F(\emptyset) = \emptyset$.

For the reverse direction, suppose that F is connected and weakly preserves pullbacks of constant maps. The product $(X_1 \times X_2, \pi_1, \pi_2)$ is at the same time the pullback of the terminal maps $!_{X_i} : X_i \to \mathbf{1}$. Applying F and considering that $F(\mathbf{1}) \cong \mathbf{1}$, we see that the $F!_{X_i}$ are also terminal maps, so their pullback is $(F(X_1) \times F(X_2), \eta_1, \eta_2)$. Thus, if F weakly preserves the pullback of the $!_{X_i}$, then we must have that $(F(X_1 \times X_2), F\pi_1, F\pi_2)$ is a weak pullback of the $F!_{X_i}$ which by Lemma 4.4 means that there exists a surjective map $\delta : F(X_1 \times X_2) \to F(X_1) \times F(X_2)$ with $\eta_i \circ \delta = F\pi_i$. □

The following example demonstrates that the requirement that F has no constants is essential in Theorem 4.7.

Example 4.8. Consider the functor T with $T(X) = X^2/\Delta$ where Δ is the equivalence relation on X^2 identifying any two elements in the diagonal of X^2. For $x_1, x_2 \in X$, we denote the elements of X^2/Δ by (x_1, x_2) if $x_1 \neq x_2$ and by $\perp$ otherwise. On maps $f : X \to Y$ the functor T is defined as $(Tf)(\perp) = \perp$ and

$$(Tf)(x_1, x_2) = \begin{cases} \perp & \text{if } f(x_1) = f(x_2), \\ (f(x_1), f(x_2)) & \text{else.} \end{cases}$$

Then T is a functor and the projection $\pi_\Delta : X^2 \to X^2/\Delta$ is a natural transformation. Even though $T(\emptyset) = \emptyset$, the functor does have a constant, $\perp$.

The map $\delta = (T\pi_1, T\pi_2) : T(X \times Y) \to T(X) \times T(Y)$ is surjective: If $X = \emptyset$ or $Y = \emptyset$ this is trivial, otherwise fix some $x \in X$ and $y \in Y$. Then $((x_1, x_2), (y_1, y_2)) \in T(X) \times T(Y)$ has preimage $((x_1, y_1), (x_2, y_2))$. Preimages of $((x_1, x_2), \perp)$ and of $(\perp, (y_1, y_2))$ are $((x_1, y)(x_2, y))$ and $((x, y_1), (x, y_2))$. Finally $(\perp, \perp)$ has preimage $\perp$. Thus T weakly preserves products.

To see that T does not weakly preserve pullbacks of constant maps, consider $c_0^X, c_1^X : X \to \{0, 1\}$ whose pullback is $\emptyset$. But $T(c_0^X) = T(c_1^X) = c_\perp^{T(X)}$ and their pullback is $T(X) \times T(X)$. Clearly there is no way to find a surjective map from $T(\emptyset) = \emptyset$ to $T(X) \times T(Y)$ as would be required by Lemma 4.4.

5. Proof of the main theorem

We are finally turning to the proof of Theorem 2.1, verifying the surjectivity of $\delta = (F\pi_1, F\pi_2)$ when (F, ι, μ) is a monad. Thus given $(p, q) \in F(A_1) \times F(A_2)$, we are required to find an element $t \in F(A_1 \times A_2)$ such that $(F\pi_1)(t) = p$ and $(F\pi_2)(t) = q$.

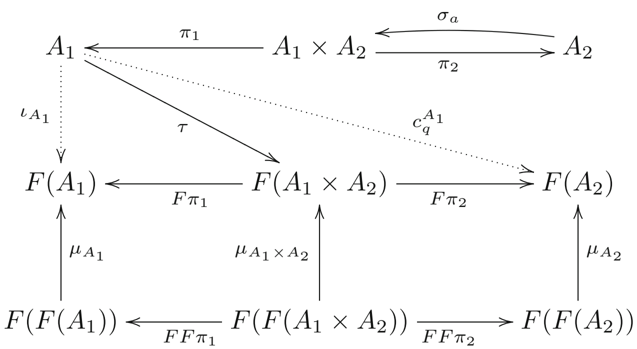
For each $a \in A_1$ we define a map $\sigma_a : A_2 \to A_1 \times A_2$ by

$$\sigma_a(b) := (a, b).$$

Next we define $\tau : A_1 \to F(A_1 \times A_2)$ by

$$\tau(a) := (F\sigma_a)(q).$$

The following picture gives an overview, where the lower squares commute due to the fact that μ is a natural transformation,



and the commutativities involving the dotted arrows will be established in the following auxiliary lemma:

Lemma 5.1.

(1) $F\pi_1 \circ \tau = \iota_{A_1}$
(2) $F\pi_2 \circ \tau = c_q^{A_1}$.

Proof. From the definition it follows that $\pi_1 \circ \sigma_a = c_a^{A_2}$ and $\pi_2 \circ \sigma_a = id_{A_2}$. Using these, and Lemma 3.1, we calculate:

$$\begin{aligned}(F\pi_1 \circ \tau)(a) &= (F\pi_1)(\tau(a)) \\ &= (F\pi_1)((F\sigma_a)(q)) \\ &= ((F\pi_1) \circ F\sigma_a)(q) \\ &= F(\pi_1 \circ \sigma_a)(q) \\ &= (Fc_a^{A_2})(q) \\ &= c_{\iota_{A_1}(a)}^{F(A_2)}(q) \\ &= \iota_{A_1}(a)\end{aligned}$$

and similarly

$$\begin{aligned}(F\pi_2 \circ \tau)(a) &= F(\pi_2)((F\sigma_a)(q)) \\ &= F(\pi_2 \circ \sigma_a)(q) \\ &= F(id_{A_2})(q) \\ &= id_{F(A_2)}(q) \\ &= q\end{aligned}$$

whence $(F\pi_2 \circ \tau)$ is the constant map $c_q^{A_1} : A_1 \to F(A_2)$. □

With these lemmas in place, we can finish the proof of Theorem 2.1. We set

$$t := (\mu_{A_1 \times A_2} \circ F\tau)(p)$$

and claim:

$$(F\pi_1)(t) = p \tag{5.1}$$

$$(F\pi_2)(t) = q. \tag{5.2}$$

In order to show (5.1), we calculate, using naturality of μ, for $i = 1, 2$:

$$\begin{aligned}(F\pi_i)(t) &= (F\pi_i)((\mu_{A_1 \times A_2} \circ F\tau)(p)) \\ &= (F\pi_i \circ \mu_{A_1 \times A_2} \circ F\tau)(p) \\ &= (\mu_{A_i} \circ FF\pi_i \circ F\tau)(p) \\ &= (\mu_{A_i} \circ F(F\pi_i \circ \tau))(p).\end{aligned}$$

Then for $i = 1$ we continue, using Lemma 5.1 and the first monad law:

$$\begin{aligned}(\mu_{A_1} \circ F(F\pi_1 \circ \tau))(p) &= (\mu_{A_1} \circ F\iota_{A_1})(p) \\ &= id_{F(A_1)}(p) \\ &= p,\end{aligned}$$

whereas for $i = 2$ we obtain, using Lemmas 5.1 and 3.1 as well as the second monad law:

$$\begin{aligned}(\mu_{A_2} \circ F(F\pi_2 \circ \tau))(p) &= (\mu_{A_2} \circ F(c_q^{A_1}))(p) \\ &= (\mu_{A_2} \circ c^{F(A_1)}_{\iota_{F(A_2)}(q)})(p) \\ &= \mu_{A_2}(\iota_{F(A_2)}(q)) \\ &= (\mu_{A_2} \circ \iota_{F(A_2)})(q) \\ &= q.\end{aligned}$$

Corollary 5.2. *Let* $\alpha = \mathrm{Ker}\,\pi_1$ *and* $\beta = \mathrm{Ker}\,\pi_2$, *then*

$$F(A \times B)/\alpha \wedge \beta \cong F(A) \times F(B).$$

6. Conclusion

We have shown that a key observation in the work of Dent, Kearnes and Szendrei [2] results from a weak limit preservation property which results from the free-algebra functor $F_{\mathcal{V}}$ being a (not necessarily associative) monad. Such weak limit preservation properties of $Set-$functors are highly relevant when using such functors as type functors for coalgebras.

Indeed, in a forthcoming paper [10] weak preservation of kernel pairs and preservation of preimages by $F_{\mathcal{V}}$ will be characterized by syntactic criteria for the equations Σ defining the variety $\mathcal{V}$.

Acknowledgements

Open Access funding provided by Projekt DEAL. I am sincerely indebted to Peter Jipsen and Andrew Moshier for inspiring discussions during my stay at Chapman University, where the main result of this note was obtained.

References

[1] Chajda, I., Czédli, G., Horváth, E.: The shifting lemma and shifting lattice identities. Algebra Universalis **50**, 51–60 (2003). https://doi.org/10.1007/s00012-003-1808-2

[2] Dent, T., Kearnes, K.A., Szendrei, Á.: An easy test for congruence modularity. Algebra Universalis **67**(4), 375–392 (2012). https://doi.org/10.1007/s00012-012-0186-z

[3] Gran, M., Rodelo, D., Nguefeu, I.T.: Variations of the shifting lemma and goursat categories. Algebra Universalis **80**(1) (Jan 2019). https://doi.org/10.1007/s00012-018-0575-z

[4] Gumm, H.P.: Elements of the general theory of coalgebras. In: LUATCS 99. Rand Afrikaans University, Johannesburg, South Africa (1999), https://www.mathematik.uni-marburg.de/~gumm/Papers/Luatcs.pdf

[5] Gumm, H.P.: Functors for coalgebras. Algebra Universalis **45**, 135–147 (2001). https://doi.org/10.1007/s00012-001-8156-x

[6] Gumm, H.P., Schröder, T.: Coalgebraic structure from weak limit preserving functors. In: Reichel, H. (ed.) Coalgebraic Methods in Computer Science. Electronic Notes in Theoretical Computer Science, vol. 33, pp. 113–133. Elsevier Science, Amsterdam (2000). https://doi.org/10.1016/S1571-0661(05)80346-9

[7] Gumm, H.P., Schröder, T.: Monoid-labeled transition systems. Electron. Notes Theor. Comput. Sci. **44**, 184–203 (2001). https://doi.org/10.1016/S1571-0661(04)80908-3

[8] Gumm, H.P., Schröder, T.: Types and coalgebraic structure. Algebra Universalis **53**, 229–252 (2005). https://doi.org/10.1007/s00012-005-1888-2

[9] Gumm, H.P.: Geometrical methods in congruence modular algebras. No. 286 in Memoirs of the AMS, American Mathematical Society (1983), https://bookstore.ams.org/memo-45-286

[10] Gumm, H.P.: Free-algebra functors from a coalgebraic perspective. In: Coalgebraic Methods in Computer Science (CMCS 2020). IFIP-LNCS, Springer (to appear), https://arxiv.org/abs/2001.08453

[11] Manes, E.G.: Implementing collection classes with monads. Math. Struct. Comput. Sci. **8**(3), 231–276 (1998). https://doi.org/10.1017/S0960129598002515

[12] Moggi, E.: Notions of computation and monads. Inf. Comput. **93**(1), 55–92 (1991). https://doi.org/10.1016/0890-5401(91)90052-4. selections from 1989 IEEE Symposium on Logic in Computer Science

[13] Rutten, J.: Universal coalgebra: A theory of systems. Tech. rep., CWI, Amsterdam (1996), CS-R9652

[14] Trnková, V.: On descriptive classification of set-functors. i. Commentationes Mathematicae Universitatis Carolinae **012**(1), 143–174 (1971), http://eudml.org/doc/16419

[15] Wadler, P.: Comprehending monads. Math. Struct. Comput. Sci. **2**(4), 461–493 (1992). https://doi.org/10.1017/S0960129500001560

H. Peter Gumm
Philipps-Universität Marburg
Marburg
Germany
e-mail: gumm@mathematik.uni-marburg.de

Received: 1 March 2019.
Accepted: 11 March 2020.