

MOTIVATION FOR THE MONADICITY THEOREMS

PATRICK STEVENS

<https://www.patrickstevens.co.uk/misc/MonadicTheorems/MonadicTheorems.pdf>

You should draw diagrams yourself throughout this document. It will be unreadable as mere symbols.

1. DEFINITIONS

Definition 1. *Given a monad \mathbb{T} on \mathcal{C} , the Eilenberg-Moore category is $\mathcal{C}^{\mathbb{T}}$, the category of all algebras (A, α) with $A \in \mathcal{C}$ and $\alpha : TA \rightarrow A$, whose arrows are “homomorphisms” $f \in \text{mor}(\mathcal{C})$ such that $(A, \alpha) \rightarrow (B, \beta)$ has $f\alpha : TA \rightarrow B$ equalling $(Tf)\beta : TA \rightarrow B$.*

Definition 2. *Given \mathbb{T} a monad induced by $(F : \mathcal{C} \rightarrow \mathcal{D}) \dashv (G : \mathcal{D} \rightarrow \mathcal{C})$, where the unit and counit of the adjunction are η, ϵ respectively, the comparison functor $K : \mathcal{D} \rightarrow \mathcal{C}^{\mathbb{T}}$ is given by $K(B) = (GB, G\epsilon_B)$ and $K(f : A \rightarrow B) = Gf$.*

Definition 3. *An adjunction $(F : \mathcal{C} \rightarrow \mathcal{D}) \dashv (G : \mathcal{D} \rightarrow \mathcal{C})$ is monadic if the comparison functor $K : \mathcal{D} \rightarrow \mathcal{C}^{\mathbb{T}}$ is part of an equivalence, where \mathbb{T} is the monad induced by the adjunction.*

That is, a monadic adjunction is one of the form “ \mathcal{C} is being taken to a category of algebras”, like the free-group/forgetful functor adjunction.

Definition 4. *A functor $G : \mathcal{D} \rightarrow \mathcal{C}$ is monadic if it has a left adjoint and the adjunction is monadic.*

2. ALGEBRAS ARE COEQUALISERS

This is a lemma we’re going to need a lot, so without motivating why, I’ll just present it. I’ll use the perhaps nonstandard terminology that α *coequalises* two maps if the composites are equal, and it is a *coequaliser* if it is universal among all coequalising maps.

Theorem 1. *Let (A, α) be an algebra for monad $T : \mathcal{C} \rightarrow \mathcal{C}$. Then $\alpha : TA \rightarrow A$ is a coequaliser.*

The way to think about this is from the free-group/forgetful-functor monad on **Sets**. In this context, it means: “every group is a quotient of a free group”.

In the free-group context, (A, α) is a set A together with a map $\alpha : TA \rightarrow A$ which “tells us how to multiply”. That is, α precisely specifies the multiplication table for the group.

Consider TTA , which is the set of words of words of elements of A . On the one hand, to get from a word $((a_1, a_2, \dots), (b_1, b_2, \dots), \dots)$ to an element of A , we could multiply out each word and then multiply again; on the other, we could flatten the word-of-words and then multiply.

(Remember, $\alpha : TA \rightarrow A$ is the multiplication, and $\mu_A : TTA \rightarrow TA$ is the flattening operation. Note that $\alpha : (TA, \mu_A) \rightarrow (A, \alpha)$ is an algebra homomorphism, because $\alpha\mu_A = \alpha(T\alpha)$ is just part of the definition of an algebra homomorphism.)

$$(TTA, \mu_{TA}) \begin{array}{c} \xrightarrow{T\alpha} \\ \xrightarrow{\mu_A} \end{array} (TA, \mu_A) \xrightarrow{\alpha} (A, \alpha)$$

I claim that doing these two gives the same result: that is, α coequalises the maps $T\alpha$ and $\mu_A : (TTA, \mu_{TA}) \rightarrow (TA, \mu_A)$. Indeed, the fact that $\alpha(T\alpha) = \alpha\mu_A : TTA \rightarrow A$ is just part of the definition of an algebra homomorphism. From the groups point of view, it’s true that we can re-group terms and then multiply, and still get the same answer as just multiplying.

I now claim that it’s actually a coequaliser: any homomorphism $f : (TA, \mu_A) \rightarrow (B, \beta)$ which has $f(T\alpha) = f\mu_A$ must factor through $\alpha : TA \rightarrow A$. From the groups point of view, if we do some operation on the words which doesn’t care about the grouping of the terms, it must be defined by its action on the generators (elements of the set A).

$$\begin{array}{ccc} (TTA, \mu_{TA}) & \begin{array}{c} \xrightarrow{T\alpha} \\ \xrightarrow{\mu_A} \end{array} & (TA, \mu_A) & \xrightarrow{\alpha} & (A, \alpha) \\ & & \searrow f & & \vdots g \\ & & & & (B, \beta) \end{array}$$

Indeed, if we apply the forgetful functor, we obtain a split coequaliser (and hence a fortiori a coequaliser), because:

- it coequalises
- $\eta_A : A \rightarrow TA$ inverts $\alpha : TA \rightarrow A$ (“inclusion into the free group” vs “multiplying out one-element words”)
- $\eta_{TA} : TA \rightarrow TTA$ inverts $T\alpha : TTA \rightarrow TA$

So there can only be one possible factorisation of f through α (because forgetting the algebra structure would keep the factorisation the same, and we’ve just shown that forgetting the algebra structure we already have a coequaliser).

There actually is a factorisation of f through α : say g is the factorisation $A \rightarrow B$ once we’ve forgotten the algebra structure. Then we just need to show that g is an algebra

homomorphism $(A, \alpha) \rightarrow (B, \beta)$. (That is, we need to show that after we've forgotten the algebra structure, we can add in the algebra structure again without breaking g .)

But it is: $g\alpha = \beta(Tg)$ because:

- We already know $g\alpha = f : TA \rightarrow B$ (by “ g is a factorisation of f through α ”)
- $f\mu_A = \beta(Tf)$: because $f : TA \rightarrow B$ is an algebra hom

From the second and first together, we have $g\alpha\mu_A = \beta(Tg)(T\alpha)$; since α is an algebra hom, we have $\alpha\mu_A = \alpha(T\alpha)$.

Therefore $g\alpha T(\alpha) = \beta(Tg)(T\alpha)$, so $fT(\alpha) = \beta(Tg)(T\alpha)$.

So it's enough for $T(\alpha)$ to be epic. Fortunately, we have already got that it's even split epic, so it is indeed epic, so we do have $g\alpha = \beta(Tg)$, so g is an algebra hom.

2.1. Summary. That is, every algebra (A, α) is a coequaliser of two free algebras, $(TTA, \mu_{TA}) \rightarrow (TA, \mu_A)$ by $T\alpha$ and μ_A .

3. OVERVIEW

Obviously we can't get rid of the need for G to have an adjoint, because “monadic” is solely a property of an adjunction. So we want to find an equivalent way to characterise the fact that the adjunction is monadic.

The “equivalence of categories” condition tells us that we somehow need to characterise when K has a left adjoint L . (Recall that every equivalence can be refined to an adjoint equivalence.)

Once we've done that, we'll need natural isomorphisms $\alpha : 1_{\mathcal{C}^{\mathbb{T}}} \rightarrow KL$ and $\beta : LK \rightarrow 1_{\mathcal{D}}$. The way we'll do that is to show that A and LKA are both coequalisers of the same thing, and (A, α) and $KL(A, \alpha)$ are also both coequalisers of the same thing. Since two objects which satisfy the UMP of the coequaliser must be isomorphic by a *unique* isomorphism, this will prove the existence (“isomorphic”) and naturality (“unique”) of the bijection. (Recall that a natural transformation is iso if and only if it is pointwise iso.)

4. ADJOINT FOR K

We're looking for $L : \mathcal{C}^{\mathbb{T}} \rightarrow \mathcal{D}$, such that $LKB = L(GB, G\epsilon_B)$ is isomorphic to B for each $B \in \mathcal{D}$.

4.1. Definition of L . Can we show, then, that $L(GB, G\epsilon_B) \cong B$?

This is a bit of a rabbit out of a hat, but remember that we've shown every B is a coequaliser (as is (B, β) for any hom β). Therefore, we could try and show that the following construction generalises.

$$GFGB \xrightarrow[\mu_B]{GF(G\epsilon_B)} GFB \xrightarrow{G\epsilon_B} B \cong? L(GB, G\epsilon_B)$$

(For the moment, there's not really any reason to suspect that this is an adjoint to K . I'm sorry. It is, though.)

Well, there's a way to generalise it so that the objects look right, anyway: just throw away the initial G from anywhere we can. Now the arrows are going to be nonsense, of

course, and now the congruence doesn't make any sense. (Replace ϵ_B by β , because now it's a general arrow.)

$$FGFB \begin{array}{c} \xrightarrow{F\beta} \\ \xrightarrow{?} \end{array} FB \xrightarrow{?} L(B, \beta)$$

What has to happen to the arrows? We had $G\epsilon_B = \beta$ before, but remember that in the monad (\mathbb{T}, η, μ) induced by the adjunction $F \dashv G$, we defined $\mu_B = G\epsilon_B$. So actually we can get rid of the G here!

$$FGFB \begin{array}{c} \xrightarrow{F\beta} \\ \xrightarrow{\epsilon_{FB}} \end{array} FB \xrightarrow{?} L(B, \beta)$$

This is a general definition of $L(B, \beta)$: as the coequaliser of $F\beta$ and ϵ_{FB} from $FGFB \rightarrow FB$.

Is it well-defined? That is, does the coequaliser exist?

Notice that $F\beta$ and ϵ_{FB} both have inverse $F\eta_B$, so we have a reflexive pair; so it's enough for \mathcal{D} to have coequalisers of reflexive pairs.

Alternatively, if we apply G to the diagram, we get:

$$TTB \begin{array}{c} \xrightarrow{T\beta} \\ \xrightarrow{G\epsilon_{FB}} \end{array} TB$$

But we've already seen that this is part of a split coequaliser diagram (the three bullet-points in Section 2), so it does have a coequaliser in \mathcal{C} ; therefore it's also, separately, enough for G to create coequalisers of G -split pairs.

4.1.1. *Summary.* We have seen that if we have either of

- (1) \mathcal{D} has coequalisers of reflexive pairs
- (2) G creates coequalisers of G -split pairs

then our putative left-adjoint L is well-defined as a function.

It is defined on objects as a coequaliser:

$$FGFB \begin{array}{c} \xrightarrow{F\beta} \\ \xrightarrow{\epsilon_{FB}} \end{array} FB \xrightarrow{l} L(B, \beta)$$

How do we define it on arrows? An entirely mechanical check tells us how:

$$\begin{array}{ccccc} FGFB & \begin{array}{c} \xrightarrow{F\beta} \\ \xrightarrow{\epsilon_{FB}} \end{array} & FB & \xrightarrow{l_B} & L(B, \beta) \\ \downarrow FGFf & & \downarrow Ff & & \vdots \\ FGFC & \begin{array}{c} \xrightarrow{F\gamma} \\ \xrightarrow{\epsilon_{FC}} \end{array} & FC & \xrightarrow{l_C} & L(C, \gamma) \end{array}$$

where the upper square and the lower square both commute (functoriality of F and naturality of ϵ , respectively).

We can then chase arrows to show that $l_C \circ Ff$ coequalises $F\beta$ and ϵ_{FB} , and so it must factor uniquely through l_B .

4.2. Functoriality of L . We've shown that L is defined, but is it a functor? It needs to:

- (1) preserve identities
- (2) preserve composition

Looking at the diagram at the end of the summary above (section 4.1.1), it's clear that if $f = 1_B$ then the induced $L(f)$ must be $1_{(B,\beta)}$.

For composition, if we just shove another copy of that diagram below itself, and join up all the arrows, it's another mechanical diagram-chase to show that the composition is preserved (using the uniqueness of the dashed arrows).

4.3. Adjointness of K and L . Finally, we get to the key bit!

We'll use my favourite definition of an adjoint: as an initial object of $((A, \alpha) \downarrow L)$ for each $(A, \alpha) \in \mathcal{C}^{\mathbb{T}}$.

The diagram we need to verify is:

$$\begin{array}{ccccc}
 FGFA & \xrightarrow[\epsilon_{FA}]{F\alpha} & FA & \longrightarrow & L(A, \alpha) & \overset{g}{\dashrightarrow} & B \\
 & & & & \uparrow i_{(A, \alpha)} & \nearrow \text{\scriptsize \textcircled{f}} & \\
 & & & & KL(A, \alpha) & \xrightarrow{Kg} & KB \\
 & & & & (A, \alpha) & &
 \end{array}$$

Since $KB = (GB, G\epsilon_B)$, the diagram becomes:

$$\begin{array}{ccccc}
 FGFA & \xrightarrow[\epsilon_{FA}]{F\alpha} & FA & \xrightarrow{l} & L(A, \alpha) & \overset{g}{\dashrightarrow} & B \\
 & & & & \uparrow i_{(A, \alpha)} & \nearrow \text{\scriptsize \textcircled{f}} & \\
 & & & & KL(A, \alpha) & \xrightarrow{Kg} & (GB, G\epsilon_B) \\
 & & & & (A, \alpha) & &
 \end{array}$$

4.3.1. Uniqueness of g . Notice that the UMP of the coequaliser guarantees uniqueness of the arrow g , if it exists, because $FA \rightarrow B$ coequalises $F\alpha$ and ϵ_{FA} so it factors *uniquely* through the coequaliser $L(A, \alpha)$.

So we just need to define $i_{(A, \alpha)}$ and to show that g exists for every f .

4.3.2. *Existence of g .* The only thing we can do with f is to view it as a map $A \rightarrow GB$ and then apply F to it, to raise it up to \mathcal{D} . And once we do that, it immediately becomes clear that we want to show that $\epsilon_B \circ Ff : FA \rightarrow B$ coequalises $F\alpha$ and ϵ_{FA} . When we've got that, $\epsilon_B \circ Ff$ will factor through $L(A, \alpha)$ to give our map g .

Since f is an algebra hom, we have $f\alpha = G(\epsilon_B) \circ (Tf) = G(\epsilon_B \circ Ff)$. Applying F obtain $F(f\alpha) = FG(\epsilon_B \circ Ff)$; if we hit this on the left with ϵ_B then we can use naturality on the right-hand side, so have

$$\epsilon_B F(f\alpha) = \epsilon_B FG(\epsilon_B \circ Ff) = \epsilon_B \circ Ff \circ \epsilon_{FA}$$

That is, for every arrow $f : (A, \alpha) \rightarrow K(B)$ there is exactly one arrow $g : L(A, \alpha) \rightarrow B$ such that $\epsilon_B \circ Ff = g \circ l : FA \rightarrow B$.

4.3.3. *Definition of i_A .* Conversely, any arrow $g : L(A, \alpha) \rightarrow B$ has a unique arrow $h = g \circ l : FA \rightarrow B$ satisfying $h(F\alpha) = h(\epsilon_{FA})$, and by the $F \dashv G$ adjunction, this corresponds to a unique arrow $\bar{h} = (Gh)\eta_A : A \rightarrow GB$. This tells us to define $i_{(A, \alpha)} : (A, \alpha) \rightarrow KL(A, \alpha)$ by $i_{(A, \alpha)} = (Gl)\eta_A$, where recall that η is the unit of the $F \dashv G$ adjunction.

$i_{(A, \alpha)}$ really is a homomorphism: have $KL(A, \alpha) = (GL(A, \alpha), G\epsilon_{L(A, \alpha)})$, so need

$$G\epsilon_{L(A, \alpha)} \circ T(i_{(A, \alpha)}) = i_{(A, \alpha)}\alpha$$

The right-hand side is $(Gl)\eta_A\alpha$, which by definition of an algebra hom is just Gl ; the left-hand side is

$$G(\epsilon_{L(A, \alpha)} \circ FGl \circ F\eta_A)$$

which by naturality of η is

$$G(l\epsilon_{FA} \circ F\eta_A)$$

which by a monad identity is just Gl .

So we have the homomorphism $i_{(A, \alpha)}$.

4.3.4. *Verify the triangle.* We have one thing left to do: to show that $Kg \circ i_{(A, \alpha)} = f$ for all $f : (A, \alpha) \rightarrow K(B)$.

We have

$$Kg \circ i_{(A, \alpha)} = Gg \circ Gl \circ \eta_A = G(\epsilon_B \circ Ff) \circ \eta_A$$

so by naturality of η , this is just f straight away.

4.4. **Unmotivated bits.** Once you have the idea of “generalise the fact that GB is a coequaliser”, the rest is basically forced, although it's taken me several hours to work out the steps. (“Forced” is not the same as “easy”!) That is the only unmotivated step in the construction of this adjoint.

5. MONADICITY THEOREMS

We've actually done most of the hard work now. We've shown that every algebra occurs as a certain coequaliser, and that under one of two conditions

- (1) \mathcal{D} has coequalisers of reflexive pairs
- (2) G creates coequalisers of G -split pairs

the comparison functor has a left adjoint which is defined as another coequaliser. I'll restate them here for convenience.

Theorem 2 (Algebra as coequaliser).

$$(TTA, \mu_{TA}) \begin{array}{c} \xrightarrow{T\alpha} \\ \xrightarrow{\mu_A} \end{array} (TA, \mu_A) \xrightarrow{\alpha} (A, \alpha)$$

Definition 5 (Adjoint L of K).

$$FGFA \begin{array}{c} \xrightarrow{F\alpha} \\ \xrightarrow{\epsilon_{FA}} \end{array} FA \xrightarrow{l} L(A, \alpha)$$

The unit of the adjunction is $i_{(A, \alpha)} = (Gl)\eta_A : (A, \alpha) \rightarrow KL(A, \alpha)$.

We also showed that if we forget the algebra structure from the coequaliser, then we get another coequaliser, this time in \mathcal{C} rather than \mathcal{C}^T ; this was because the algebra-coequaliser diagram is in fact a split coequaliser, so is preserved by all functors.

Remembering that $\mu_A = G\epsilon_{FA}$, we see that actually these two diagrams are extremely similar. (A, α) is a coequaliser of the diagram you get when you apply G to the diagram for $L(A, \alpha)$.

We want to find two natural isomorphisms $\gamma : 1_{\mathcal{C}^T} \rightarrow KL$ and $\delta : 1_{\mathcal{D}} \rightarrow LK$; this will demonstrate the equivalence of the two categories.

5.1. **Finding $\delta : 1_{\mathcal{D}} \rightarrow LK$.** Recall that for $B \in \mathcal{D}$, have $LK(B) = L(GB, G\epsilon_B)$. That is, $LK(B)$ is the coequaliser in the following diagram:

$$FGFGB \begin{array}{c} \xrightarrow{FG\epsilon_B} \\ \xrightarrow{\epsilon_{FGB}} \end{array} FGB \xrightarrow{l} L(GB, G\epsilon_B)$$

Now observe that $\epsilon_B : FGB \rightarrow B$ coequalises that diagram too - that's just what it means for ϵ to be natural - so it must factor through the coequaliser.

$$\begin{array}{ccc} FGFGB \begin{array}{c} \xrightarrow{FG\epsilon_B} \\ \xrightarrow{\epsilon_{FGB}} \end{array} FGB & \xrightarrow{l} & L(GB, G\epsilon_B) = LKB \\ & \searrow \epsilon_B & \downarrow \nu_B \\ & & B \end{array}$$

We want ν_B to be an isomorphism.

We had two conditions for L being well-defined.

5.1.1. *G creates coequalisers of G -split pairs.* In this case, in fact ϵ_B is a coequaliser, not just coequalising, because when we apply G to the diagram we get a coequaliser of a split pair. So immediately ν_B is an iso.

5.1.2. \mathcal{D} has coequalisers of reflexive pairs. In this case, applying G to the diagram, and assuming that G preserves coequalisers of reflexive pairs, we will get a diagram as follows, where the top line is a coequaliser:

$$\begin{array}{ccccc}
 GF GFGB & \xrightarrow[\text{\scriptsize } G\epsilon_{FGB}]{\text{\scriptsize } GF G\epsilon_B} & GFGB & \xrightarrow{\text{\scriptsize } Gl} & GL(GB, G\epsilon_B) = GLKB \\
 & & & \searrow \text{\scriptsize } G\epsilon_B & \downarrow \text{\scriptsize } G\nu_B \\
 & & & & GB
 \end{array}$$

But notice that the following is a split coequaliser diagram, and hence is a coequaliser:

$$\begin{array}{ccccc}
 & \xrightarrow{\text{\scriptsize } GF G\epsilon_B} & & & \\
 GF GFGB & \xrightarrow[\text{\scriptsize } G\epsilon_{FGB}]{\text{\scriptsize } GF G\epsilon_B} & GFGB & \xleftarrow[\text{\scriptsize } G\epsilon_B]{\text{\scriptsize } \eta_{GB}} & GB \\
 & \xleftarrow[\text{\scriptsize } \eta_{GFGB}]{\text{\scriptsize } G\epsilon_{FGB}} & & &
 \end{array}$$

Indeed:

- The rightwards maps are coequalising (because if we remove a G from the diagram, we get something which was coequalising)
- $(G\epsilon_B)\eta_{GB} = 1_{GB}$ (adjunction identity)
- $\eta_{GB}(G\epsilon_B) = (GF G\epsilon_B)(\eta_{GFGB})$ (naturality of η)
- $(G\epsilon_{FGB})(\eta_{GFGB}) = 1_{GFGB}$ (adjunction identity)

Therefore $G\nu_B$ is an isomorphism. If we make the further assumption that G reflects isos, then ν_B must be an isomorphism.

What else did we assume in this section?

- (1) G preserves coequalisers of reflexive pairs
- (2) G reflects isomorphisms

5.1.3. *Naturality of the isomorphism.* I claim that actually ν is the counit of the adjunction. This would make it automatically natural.

Recall from the start of this section (5.1) that ν_B is just the factorisation of ϵ_B across the coequaliser $l : FGB \rightarrow L(GB, G\epsilon_B)$.

Therefore it corresponds to ϵ_B in the natural bijection between “maps $FGB \rightarrow B$ which coequalise $FG\epsilon_B$ and ϵ_{FGB} ” and “maps $L(GB, G\epsilon_B)$ ”.

But ϵ_B corresponds naturally to the identity 1_{GB} via the adjunction $F \dashv G$.

So using U for the forgetful functor, $U(1_{KB}) = 1_{GB} \leftrightarrow \epsilon_B \leftrightarrow \nu_B$, all naturally, and so ν_B must be the lift of 1_{KB} over the adjunction $L \vdash K$.

Therefore ν is actually the counit.

5.2. **Finding** $\gamma : 1_{\mathcal{C}^T} \rightarrow KL$. Recall that the unit of the $L \dashv K$ adjunction was $i_{(A,\alpha)} = (Gl)\eta_A : (A, \alpha) \rightarrow KL(A, \alpha)$.

We’ll show that this unit is a natural isomorphism.

Recall that $\alpha : (TA, \mu_A) \rightarrow (A, \alpha)$ is a coequaliser; that is precisely $\alpha : KFA \rightarrow (A, \alpha)$.

Applying K to the diagram of Definition 5, we must still get something which coequalises, and so it factors through the coequaliser:

$$\begin{array}{ccc}
KFGFA & \xrightarrow[\quad K\epsilon_{FA}]{\quad KF\alpha} & KFA \xrightarrow{\quad \alpha} (A, \alpha) \\
& & \searrow \scriptstyle Kl \quad \downarrow \text{dashed} \\
& & KL(A, \alpha)
\end{array}$$

I claim that $i_{(A,\alpha)}$ is this dashed arrow. Indeed, $i_{(A,\alpha)} \circ \alpha = (Gl)\eta_A \circ \alpha = Gl = Kl$, so by uniqueness of the lift, it must be $i_{(A,\alpha)}$.

It is therefore immediately a natural transformation, being the unit of an adjunction.

It is an iso: if we forget the algebra structure, we end up with the following diagram:

$$\begin{array}{ccc}
GFGFA & \xrightarrow[\quad G\epsilon_{FA}]{\quad GF\alpha} & GFA \xrightarrow{\quad \alpha} A \\
& & \searrow \scriptstyle Gl \quad \downarrow \text{dashed} \\
& & GL(A, \alpha)
\end{array}$$

But α is still a coequaliser here (we proved earlier that it remained a coequaliser after forgetting the algebra structure), and Gl is also a coequaliser because of our two assumptions: either G preserves coequalisers of reflexive pairs (and hence the coequaliser structure of Definition 5's diagram), or else it creates coequalisers of G -split pairs and so we're immediately done.

Therefore i_A must be an isomorphism in \mathcal{C} , because α and Gl are both coequalisers.

It is also an isomorphism in $\mathcal{C}^{\mathbb{T}}$, because the forgetful functor reflects isomorphisms: suppose $f : (A, \alpha) \rightarrow (B, \beta)$ has an inverse $g : B \rightarrow A$ once the algebra structure is forgotten. Then g is an algebra homomorphism, because we just need $g\beta = \alpha T(g)$; but that is equivalent to $\beta T(f) = f\alpha$ by pre-composing f and post-composing $T(f)$. Because f is already an algebra homomorphism, this condition is true.

Therefore i_A is an iso in $\mathcal{C}^{\mathbb{T}}$, and so we are finished.

6. SUMMARY

We have shown that $K : \mathcal{D} \rightarrow \mathcal{C}^{\mathbb{T}}$ has a left adjoint L under one of two conditions; that the adjunction's unit is an isomorphism under a further assumption; and that the adjunction's counit is an isomorphism under a further assumption.

Together, this tells us that K is part of an equivalence: that is, G is monadic.

Theorem 3 (Monadicity Theorems, one direction). *Suppose $G : \mathcal{D} \rightarrow \mathcal{C}$ is a functor which has a left adjoint. If*

- (1) *G creates coequalisers of G -split pairs, or*
- (2) *G reflects isomorphisms, \mathcal{D} has coequalisers of reflexive pairs, and G preserves them*

then G is monadic.