MIDWEST TOPOLOGY SEMINAR, MARCH 10 2012

GABRIEL C. DRUMMOND-COLE

1. Emily Riehl: Lifting properties and the small object argument

Thanks very much, I want to thank the organizers. I'm excited to start us off. Today I'm going to talk about the algebraic perspective on fibrations in homotopy theory. This is a talk in three acts, the first of which is the longest. I want to talk about the interplay between lifting properties and factorizations, maybe weak factorization systems or things that spring up naturally. I'll focus on particular examples, but this is all formal so it will be true in much more generality.

1.1. **Act I.**

Let's say that we have a goal to replace a map by a fibration, let's say I'm talking about compactly generated weak Hausdorff spaces, with Serre fibrations, a map is a fibration if it satisfies the lifting property

$$\begin{tikzcd} D^n \arrow[r] \arrow[d] & X \arrow[d, "f"] \\ D^n \times I \arrow[r] \arrow[ur, dotted] & Y \end{tikzcd}$$

So what we can do, let me draw this square again and push out.

$$\begin{tikzcd} D^n \arrow[r] \arrow[d] & X \arrow[r, equal] \arrow[d, "Lf"] & X \arrow[d] \\ D^n \times I \arrow[r] \arrow[ur, dotted] & Ef \arrow[r, "Rf"] & Y \end{tikzcd}$$

I need to solve all lifting properties, so I should do this over all possible lifting problems. Now there are more lifting properties in Ef so we haven't solved this problem.

Even though we've not made any progress, this data helps us detect fibrations. If f were a fibration, we'd have a lift in the right square, if we had a lift there, then we'd get a lift in the bigger rectangle. Then we've shown that a map is a fibration if and only if there is a lift for this single lifting problem:

$$\begin{tikzcd} {} \arrow[r, equal] \arrow[d, "Lf"] & {} \arrow[d, "f"] \\ {} \arrow[r, "Rf"'] & {} \end{tikzcd}$$

This is the same thing as realizing Rf as a retract and this is the same thing to say that f is an R-algebra for the functor R on the arrow category which takes a map to its right factor.

We can say, flipping around,

Definition 1.1. *A map is an L-coalgebra if and only if there is a lift*

$$\begin{tikzcd} X \arrow[r, "Lf"] \arrow[d, "f"'] & Ef \arrow[d, "Rf"] \\ Y \arrow[r, equal] \arrow[ur, dotted] & Y \end{tikzcd}$$

A map is an L-coalgebra implies it is a trivial cofibration. An L-coalgebra has the lifting property with respect to a R-algebra. Knowing that something is a coalgebra proves that it's a trivial cofibration.

I've made no progress toward my goal. I want to bring in another example, with the Hurewicz fibrations, they should satisfy the homotopy lifting property for maps from any space.

$$\begin{tikzcd} A \arrow[r] \arrow[d] & X \arrow[d, "f"] \\ A \times I \arrow[r] \arrow[ur, dotted] & Y \end{tikzcd}$$

We can't do the thing that we said before. We can't even do the thing that didn't work before. You can't attach all homotopies. I don't know who this is due to, there's a represented functor, a contravariant functor $Top^{op} \to Set$ which takes A to squares of the inclusion $A \to A \times I$ and Y, this is represented by a space Nf. A map to the diagram

$$\begin{tikzcd} & Y^I \arrow[d] \\ X \arrow[r] & Y \end{tikzcd}$$

corresponds to a map Nf. So the map $Nf \to Nf$ corresponds to a map and we get a diagram

$$\begin{tikzcd} Nf \arrow[r] \arrow[d] & X \arrow[r, equal] \arrow[d] & X \arrow[d] \\ Nf \times I \arrow[r] \arrow[rr, bend right] & Ef \arrow[r, "Rf"'] & Y \end{tikzcd}$$

and a lemma: a map is a Hurewicz fibration if and only if it's an algebra over this R functor. [fast proof].

Now is time we should make some progress toward the original problem. Whichever kind I mean, I've encoded these as algebras for a pointed endofunctor. What might we do? An idea: iterate these factorizations. I do want to differentiate between these two examples. Let's talk about the first one. Let's do this multiple times, factor, factor it again, countably infinitely many times, I've just done Quillen's small object argument, by compactness a disk that goes into the colimit lands in some finite stage, and so the Serre fibration we construct you might call $R^\omega f$. There is redundancy in this construction. We'll attach a homotopy every single time to a disk in X. I'll have more to say about that in just a second. For the Hurewicz fibration case this doen't actually work. We could apply our functor again and again. You might hope this glues in enough homotopies. There's actually a really interesting subtlety. We'll prove this factorization and get one that works in the other context at one stroke.

There's the notion of an (algebraically) free monad on a pointed endofunctor, and what I'll take as the definition, algebras for the monad the category is isomorphic to algebras for the pointed endofunctor over whatever (spaces in this case). So F algebras are fibrations. If I can produce the free monad, its algebras will be fibrations. How do we produce one?

Let's try and build F. Why am I talking about monads? We have this notion of free algebras. Anything that is F of something, that's an F-algebra and so that's a fibration. The first thing to think of, we have $1 \to R \to R^2 \to \cdots \to R^\omega = F$, and now we have a multiplication, maybe this will work, and this does work in some cases, for pointed endofunctors. This isn't great, I have two maps $R \to R^2$, if those two maps are the same, this is a well-pointed endofunctor, and in that case R^ω is the free monad. If they're different I have to do something a little more refined.

I want to point out the difference in our two examples. This'll show why the naive thing in the Hurewicz case doesn't work. For the Serre fibrations, we have these disks $D^n \to X$, and these gave us homotopies $D^n \times I \to Ef$, and ηR takes this to $D^n \times I \to Ef \to ERf$, we can also get $D^n \to X \to^{Lf} Ef$, and so we could get $D^n \times I \to ERf$, and this is $R\eta$.

For Hurewicz fibrations, $Ef = Nf \times I \cup X$, and $ERf = NRf \times I \cup Ef$, which is $(NRf \times I) \cup (Nf \times I) \cup Y$. We get a path in Y and a point in its fiber, a path can include in two places, two different embeddings, and we have a problem at ∞.

How do we solve this problem? The actual free monad function? We'll guess in step one that it's R, and our second guesss will be F_2, the coequalizer of these two maps $R \to R^2$. There's something more complicated at step three, and if the procsess converges, that's the free monad.

Theorem 1.1. *(Garner) If we're permitted the small object argument (SOA) then this process converges and F-algebras are isomorphic to R-algebras, and you get a functorial factorization $\xrightarrow{C}\xrightarrow{F}$ called an algebraic weak factorization system (awfs) where the left factor is a comonad.*

This is a great thing! It solves the problem. The right factor is a fibration, you might worry that the right factor is not a cofibration anymore. This does, well, the left factor being a comonad is a coalgebra for itself, and coalgebras for the left factor, as I said, always lift against algebras for the right factor.

This solves the problem. Good. I meant to say, a lot of this story will be about producing better factorizations when you had something to start with. It's essential that you have something with lifting properties, that trivial cofibrations should be the things with lifting against the fibrations.

Richard Williamson first brought the issues with Hurweciz fibrations to my attention, and everything about that is joint work with Tobi Barthel who is here. For the Hurewicz case, we get a factorization into C followed by F, and we need to worry that C is not a cofibration.

Lemma 1.1. *(Garner)*
awfs $(\mathbb{C}, \mathbb{F})$ is a monad $\mathbb{F}$ together with a vertical composition law on $\mathbb{F}$-algebras.

Theorem 1.2. *(Barthel, R.)*
This works for the Hurewicz case.

We don't need to worry, without knowing any point set topology, this was a cofibration.

If we're only talking about point set, this is easy. You only need one factorization to get an h-model structure on a Top-bicomplete category that admits the small object argument. These are quite general examples.

[What about Cole?]

[Peter May: Cole made a mistake. Now you have the h-structure everywhere and so you have the m-structure everywhere which is where you really want to work.]

1.2. **Act II.** Often you have factorizations. We produced one that we needed and didn't have by other means. We may be able to make factorizations that we already have better. A corollary is that in any cofibrantly generated model category that permits the small object argument, there exists a cofibrant replacement comonad and a fibrant replacement monad, maybe we're talking about simplicial sets, you can get something that's a monad, and you can do even better, that's the point of Act II. I'll specialize to the case of a simplicial model category permitting the small object argument. If you prefer another enrichment, tensoring with base objects should be a left Quillen functor. That's two thirds of SM7. It's true in this setting but I don't need quite this much. The functor sending f to step 0 of the small object argument, the coproduct over disks, this is not a simplicial functor, not simplicially enriched. We've enriched the lifting properties, but not the factorizations at all. This is why. Our factorizations are already broken. There's something else we could try to do instead. The squares from j to f is just a set. But I can talk about the object $\underline{Sq}(j, f)$, which is the pullback of

$$\begin{tikzcd} & \underline{M}(dom\ j, dom\ f) \arrow[d] \\ \underline{M}(cod\ j, cod\ f) \arrow[r] & \underline{M}(dom\ j, cod\ f) \end{tikzcd}$$

So now if we use $\underline{Sq}$ we get a simplicial functor, and the underling set is correct.

Theorem 1.3. *In this context you get simplicially enriched factorizations and also fibrant and cofibrant replacement.*

This is true with the modification, if you ran Quillen's small object argument, you don't know that they factor where you want. If you run Garner's algebraic version, you do have control, and it's easy to see that the factorizations at the end are the right thing. I don't know what happens at the end, it's this complicated formula. The factorizations we see now, you take the coproduct over n of $\underline{Sq}(j, f) \otimes j$. If f is a fibration, you get a lift as in Act I. At this stage, Rf is not a fibration, but R-algebras are exactly fibrations. Because we can recognize the algebras as fibrations, we can use this free monad thing. It's also possible to check long-windedly, but the left factor lifts against the right factor so it's a cofibration.

1.3. **Act III.** I want to shift gears and talk about cofibrations. Fibrations are algebras for some monad. Every fibration is an algebra for some monad. I also have coalgebras for a comonad. Let me pretend I said trivial fibrations. It's not actually true that all cofibrations are coalgebras for the comonad. They are for the pointed endofunctor. Being a coalgebra for the comonad is a little more subtle, these are called cellular cofibrations. For example, taking the sphere inclusions, you get relative cell complexes. I guess the point of this digression is that if you want to pay special attention to the relative cell complexes, this makes it easy to do so. We have really good closure properties. Let me start by proving this. The comonad coalgebras are closed under pushout, composition, and the forgetful functor to the arrow category in Top creates all colimits. A coequalizer of cofibrations need not be a cofibration in general, but if everything is cellular and your maps preserve the structure, then the colimits are coalgebras and hence cellular cofibrations.

For a long time we have hypothesized (Richard Garner and I) that these coalgebras are exactly the cellular cofibrations.

Theorem 1.4. *T. Athorne. An algebraic relative cell complex is the same thing as a coalgebra over the comonad.*

A stratum is a space X together with a set of cells togethre with attaching maps. We can push this out, and X is the "boundary" of the stratum and $\underline{X}$, the pushout, is the body of the stratum.

What's an alegebraic notion of a relative cell complex? It's a connected sequence (the body of one is the boundary of the next) that is *proper*, meaning the attaching maps don't attach in previous strata.

There's a forgetful functor from Strata to Top which has a right adjoint which I will call L. I can look at all squares, all cells in the domain of X,

$$\begin{tikzcd}
S^{n-1} \arrow[r] \arrow[d] & X \arrow[r, equal] \arrow[d, "Lf"] & X \arrow[d, "f"] \\
D^n \arrow[r] & Ef \arrow[r, "Rf"'] & Y
\end{tikzcd}$$

How do we construct this? We could do an iterative process, this gives something that isn't proper. To construct the adjoint, you use the algebraic small object argument, and get the free monad construction again. This has a composition and so you get the comonad structure.