CGP DERIVED SEMINAR

GABRIEL C. DRUMMOND-COLE

1. Feb 19: Damien Lejay, T-structures

We'll review the modern theory about T-structures. I won't give any proofs. They are either evident or require computations I don't want to make.

I'll try to use the diagrammatic way of thinking, rectangles more than triangles.

So I won't be precise, and I want somehow to settle a bit of vocabulary, we have been talking about triangulated categories, but what we really want is stable ∞-categories, which are gadgets with certain properties.

An ∞-category has objects, for every two objects it has a homotopy type of maps between them, so you can think of this as a topological space up to weak equivalence. You see the problem of saying that I have maps. I don't want to say how to compose maps because it should be homotopical.

This is a regular ∞-category, instead of having a set of maps, you have a space of maps, and from this you can always make a regular category by taking the same objects, and the maps are the connected components of the maps from x to y. This is called the homotopy category.

Here I'm being very loose because once the language is set up I can just use the language. Then I can kill the homotopy information and just get the category. Now I want it to be stable, so I add a 0 object, coproducts, which are the same as products (finite ones). Because of my zero object, I have a map from $X \coprod Y \to X \prod Y$ which is an equivalence. I also want to have pushouts and pullbacks. So for instance the pushout of $X \xrightarrow{f} Y$ along $X \to 0$ is $\mathrm{Cof}(f)$ and the pullback of f along $0 \to Y$ is $\mathrm{Fib}(f)$. I use the special notation ΣX for the pushout $0 \leftarrow X \to 0$ and ΩX for the pullback $0 \to X \leftarrow 0$, and I want stability, Σ is the inverse of Ω.

These axioms are very strong, and there are many ways to see them, in different settings. An equivalent description, is that a square

$$\begin{tikzcd} X \arrow[r] \arrow[d] & Y \arrow[d] \\ Z \arrow[r] & T \end{tikzcd}$$

is a pushout if and only if it is a pullback.

So for $Y = Z = 0$, that says that $\Omega(\Sigma X) \cong X$. I drew the pushout but I can do the other order.

The key thing is that when $\mathcal{C}$ is stable, the homotopy category is canonically triangulated. I think it's not true that every triangulated category is the homotopy category of a stable ∞-category but they are artificial.

Let me remind you of the octahedral axiom that is easy to get back in this context. What people call a distinguished triangle is a cofiber

$$\begin{tikzcd}
X \arrow[r, "f"] \arrow[d] & Y \arrow[d] \\
0 \arrow[r] & \mathrm{Cof}(f)
\end{tikzcd}$$

and then the thing you do is compose two maps.

$$\begin{tikzcd}
X \arrow[r, "f"] \arrow[d] & Y \arrow[r, "g"] \arrow[d] & Z \\
0 \arrow[r] \arrow[d] & \mathrm{Cof}(f) \arrow[r] \arrow[d] & \mathrm{Cof}(g \circ f) \arrow[d] \\
0 \arrow[r] & 0 \arrow[r] & \mathrm{Cof}(g)
\end{tikzcd}$$

By abstract nonsense, the bottom right triangle is a pushout, so it's a distinguished triangle, and this is the octahedral axiom.

Here I'll try not to say triangulated but stable.

I recall examples of stable categories, spectra, the derived category of a nice enough Abelian category. Once you have some building blocks, the traditional category theory language lets you do this, you can for instance take categories of sheaves, sheaves of chain complexes, of spectra, all the things you could do in the Abelian setting you can do in the stable setting.

The only problem with triangulated and stable categories is that there is an *anti-theorem.*

Theorem 1.1. *If $\mathcal{C}$ is stable and $X \to Y$ is a monomorphism (think of a subobject), then $X \cong Y$.*

You don't have monomorphisms. The same is true for epimorphisms. So you can't factorize through images. Somehow all your tools, your gadgets are broken, so you need another way to factorize your maps, and this is where you introduce T-structures.

Normally this is given in your category, so you need to add data in how you factor or truncate. You need to rewire your mind because your intuition changes things. The T-structure replaces the intuition.

If you have $\mathcal{C}$ stable you can define a T-structure either on $\mathcal{C}$ or on $h\mathcal{C}$, you give yourself full subcategories $\mathcal{C}_{\geq 0}$ and $\mathcal{C}_{0\geq}$. The axioms are:

(1) You want $\mathcal{C}_{\geq 0}[1] \subset \mathcal{C}_{\geq 0}$,
(2) you want $\mathrm{Hom}(x, y) = 0$ if $x \in \mathcal{C}_{\geq 0}$ and $y \in \mathcal{C}_{-1\geq}$, and
(3) you want a factorization, a fiber/cofiber sequence $\tau_{\geq 0} X \to X \to \tau_{-1\geq} X$.

Now you can resplit your things.

One comment I made last week is that there is too much data here. One thing is that you only need to know $\mathcal{C}_{\geq 0}$, because you can get back the other one, $\mathcal{C}_{0\geq}$ is the objects x such that $\mathrm{Hom}(x, y) = 0$ for $y \in \mathcal{C}_{\geq 1}$.

These two are very nice. For example, the category $\mathcal{C}_{0\geq}$ is a localization, and $\mathcal{C}_{\geq 0}$ is a colocalization, a coreflective subcategory.

Remember that the heart is the intersection of $\mathcal{C}_{0\geq}$ and $\mathcal{C}_{\geq 0}$ and this is always an Abelian category. If you compose truncations, you have truncations that go to the heart. In this you can call this π_0 or H_0, if you use a shift, this lets you define π_n for $n \in \mathbb{Z}$. You have the fact that the heart is quivalent to $\mathcal{C}_{n\geq} \cap \mathcal{C}_{\geq n}$, and this is how you get your H_n or π_n. If you don't give yourself a T-structure, then you can't get those things. Then the important thing is the long exact sequences. If you give yourself a fiber sequence, then by computing these homotopy groups you get a long exact sequence, and that's absolutely impossible to do unless you have a T-structure.

So what I wanted to do is show you how you can forget about one of the two subcategories.

We said that $\mathcal{C}_{0\geq}$ is a localization of $\mathcal{C}$, so when you have such a thing, a reflective subcategory, you always have the class S of maps in $\mathcal{C}$ such that $L(S)$ becomes an isomorphism. You look at maps that become isomorphisms after localization, and this is equivalent, so that $\mathcal{C}_{0\geq}$ is the same as $\mathcal{C}[S^{-1}]$. When you have a T-structure, you have in particular a localization, I'll now say that this is a special class of maps that has properties:

- every isomorphism is in S,
- if $h = g \circ f$, I have the two out of three property, meaning if any two maps are in S then the third one is in S. This is trivial, since it's something true of isomorphisms.
- The third condition (this is not typical) is that S should be pushout stable. The pushout of an S-map is in S. That's the property that this class S satisfies.

Call a class satisfying these three conditions (*quasi-saturated*) then the following things are

Now I'll say how you generate a T-structure.

Proposition 1.1. *Let $\mathcal{C}$ be stable, equipped with a localization, and suppose the localization is by a class of maps S that are quasi-saturated. Then the following are equivalent:*

(1) *There exist a class of maps $f : 0 \to X$ which generate S (S is the smallest quasi-saturated class containing those maps),*
(2) $\mathcal{C}_{ge0} = \{A | LA = 0\}$ *and* $\mathcal{C}_{-1\geq} = \{A | LA = A\}$

So you can write $\mathcal{C}^+$ as

$$\bigcup_n \mathcal{C}_{n\geq}$$

which is the subcategory of left-bounded objects. In particular if $\mathcal{C}$ is $\mathcal{C}^+$ then $\mathcal{C}$ is left-bounded. Similarly you can make a subcategory of right-bounded objects $\mathcal{C}^-$ and say that $\mathcal{C}$ is right-bounded if $\mathcal{C} = \mathcal{C}^-$.

When you have a stable category with a T-structure you want to figure out the long exact sequence. But you want a recognition principle that tells me something about going back from the information of the π_n. You do something by induction, proving n by n that something is an isomorphism or something is zero. So what you want is to be able to recover your full object from its truncations.

You say that $\mathcal{C}$ is *left t-complete* if

$$\mathcal{C} \to \lim_n \mathcal{C}_{n\geq}$$

is an equivalence. You say that $\mathcal{C}$ is *right t-complete* if

$$\mathcal{C} \to \lim_n \mathcal{C}_{\geq n}$$

is an equivalence.

In any case you call $\lim_n \mathcal{C}_{n\geq}$ is $\hat{\mathcal{C}}$. All the examples you know are already right t-complete and in nature the question is whether it's left t-complete. So $\hat{\mathcal{C}}$ is a stable ∞-category and always left complete.

Let me give an example, the category of spectra is both left and right t-complete.

You get a map from $\mathcal{C}^+$ to $\mathcal{C}$ and this gives a map $\widehat{\mathcal{C}^+} \to \hat{\mathcal{C}}$, which is always an equivalence, and you have an equivalence between the left bounded categories and the left t-complete categories, via $\mathcal{C} \mapsto \mathcal{C}^+$ and $\mathcal{D} \mapsto \hat{\mathcal{D}}$.

So I want $\mathcal{C}_{\geq 0}$ to be stable under countable products, and if you have this, and the intersection of $\mathcal{C}_{\geq n}$ is zero, then $\mathcal{C}$ is left complete.

So a non-example, $\mathcal{D}(A)$ is maybe not left t-complete. You take $\mathcal{A}$ to be $\mathbb{G}_a$-representations over $\mathbb{F}_p$.

You can take $\prod_{\geq 1} A[n]$, each component has nothing in degree zero, and H_0 of that can be non-zero.

If A is Grothendieck then $\mathcal{D}(A)$ is right-complete. This satisfies AB5 and so most derived categories are right complete. Then the question is about left completeness. Let's have a break and then I'll talk more.

If I take $\mathcal{D}(A)^-$, the bounded (below) derived category of an Abelian category, then this is always left complete. If A is nice enough, and I take $\mathcal{C}$ a left-complete stable infinity category with a t-structure, and I have an exact functor from A to the heart of $\mathcal{C}$. Then you can extend to a map $\mathcal{D}(A)^- \to \mathcal{C}$ which is t-exact.

What does t-exact mean? A functor between categories with t-structures is t-exact if it sends $\mathcal{C}_{\geq 0}$ to $\mathcal{D}_{\geq 0}$ and $\mathcal{C}_{0\geq}$ o $\mathcal{D}_{0\geq}$.

This is the universal property of this $\mathcal{D}(A)^-$. There is a substatement where things are right exact you can look up. This is something that makes you wish for completeness of a category, you get a map from $\mathcal{D}(\mathcal{C}^\heartsuit)^-$ to $\mathcal{C}$, and usually you have enough injectives, so

$$\mathcal{D}(\mathcal{C}^\heartsuit)^+ \to \mathcal{C}$$

where $\mathcal{C}$ is right complete and $\mathcal{C}^\heartsuit$ has enough injectives.

You take a stable category with a t-structure. You have a comparison map.

So now if I have F a right exact functor from A to B, and what people usually do is take the derived functors, So now if you take $(\mathcal{D}(A)^-)^\heartsuit$ that's A, and so this is already a functor between hearts, and this gives a functor $\mathcal{D}^-(A) \to \mathcal{D}^-(B)$, and this is the derived functor $\mathbb{L}F$ of F (the *left*-derived functor). This is only right t-exact even though it preserves all small limits and colimits.

So as an example, derive the tensor product, you take the category of Abelian groups and tensor by $\mathbb{Z}_2$. You get a derived functor $\mathcal{D}(\mathrm{Ab})^- \to \mathcal{D}(\mathrm{Ab})^-$ which is $\otimes^{\mathbb{L}}_{\mathbb{Z}_2}$, and this will be exact and right t-exact. You can just reverse everything and you can reverse everything. So if you want to derive the global section functor of sheaves, this is left but not right exact, so you get something similar.

Let me make a remark for people who might recognize something. The completion issues, for derived functors it's well-known how to make derived functors.

If you do a bit of derived algebraic geometry, you're going to consider not just varieties but schemes and derived schemes and derived stacks, and something that people want to define are quasi-coherent sheaves over X. Sometimes this is not the object you want. Sometimes people will derive "ind-coherent" sheaves. The quasi-coherent sheaves is left t-complete and ind-coherent sheaves are not, and this map is just the completion. If X is a smooth variety then ind-coherent sheaves of X is already left complete, so these coincide. Most things are right complete, and that left completeness is the key difference.

If you remember correctly, we want to study motives, general cohomology theories on algebraic varieties. If you take the topological version of that, it's spectra, and the key property is that this is stable and has a t-structure. When people look for something motivic, they're looking for triangulated structures, and when we talk about pure motives, that's the heart of the t-structure, and this is the modern approach to all of that. So this is what people have in mind, at least as a goal.

A last comment for five minutes, stable ∞-categories plus t-structures is the input data to be able to speak of spectral sequences. If you've ever heard of spectral sequences, where can I compute these? a stable ∞-category and a t-structure. That's the typical place to do this. The goal here is to compute the π_n of a directed colimit of X_p, and then there's a recipe using the π_n of all the cofibers, and this has a modern interpretation in stable ∞-categories with a t-structure.

Every time you have a spectral sequence, usually you can rephrase it in something like this. The takehome is that a stable ∞-category without its t-structure is useless, and the t-structure is something very categorical that has a lot of implications, and in examples there are t-structures that are not obvious. It's not, yeah, I see my t-structures, they're obvious, but you have some subtle issues. Next week we have the pleasure of hearing Rune talking about a cohomology theory that people can compute on varieties.