# Limits in the category of categories

Joseph Hua

February 3, 2022

## 1 Introduction

In this project, I show that `Cat.u u`, the category of small categories is complete, where "small" in `lean` is taken to mean the universe levels (sizes) for objects and morphisms is the same. My work can easily be adapted to show that relaxing the "small" condition, `Cat.v u` has all finite limits. Since the category of categories is a bicategory, and it is quite rare in practice to work with strict equality of functors, the "morally correct" limits to consider should really be 2-limits, where the functors are considered up to natural isomorphism. However, these are not considered here, and existence of 1-limits holds anyway, which should deduce the 2-limit case.

An obvious next step is to show results for colimits, which I have not yet done.

## 2 Mathlib Results

The category theory library in `mathlib` already contains definitions for limits, using cones over diagrams.

> **Definition – Diagrams, cones and limits**
>
> A diagram in a category $\mathcal{C}$ is simply a functor from some indexing category $F : \mathcal{I} \to \mathcal{C}$. Diagrams are not explicitely defined in `lean`, as this would lead to extra unnecessary definitional equalities.
>
> $$\begin{tikzcd} i \arrow[r, bend left, "\psi"] \arrow[r, bend right, "\phi"'] & j & {} \arrow[r, dotted, "\mathcal{F}"] & {} & \mathcal{F}i \arrow[r, bend left, "\mathcal{F}\psi"] \arrow[r, bend right, "\mathcal{F}\phi"'] & \mathcal{F}j \end{tikzcd}$$
>
> A diagram for equalizers
>
> A cone over a diagram $F : \mathcal{J} \to \mathcal{C}$ consists of an apex $X$ an object in $\mathcal{C}$ and a natural transformation from (the constant functor at) $X$ to $F$. This natural transformation captures the data of legs from $X$ into the diagram that commute with the maps in the diagram.
>
> ```
> structure cone (F : J ⥤ C) :=
> (X : C) (π : (const J).obj X ⟶ F)
> ```
>
> Here `lean` recognizes that `F` is a functor, so it infers that the categorical map into `F` should be in the functor fategory, i.e. a natural transformation.

$$\begin{tikzcd}
 & X \arrow[dl, "\pi\ i"'] \arrow[dr, "\pi\ j"] & \\
\mathcal{F}i \arrow[rr, bend left, "\mathcal{F}\psi"] \arrow[rr, bend right, "\mathcal{F}\phi"'] & & \mathcal{F}j
\end{tikzcd}$$

A cone over the equalizer diagram, or a "fork"

Finally, the data of a cone $l$ being the limit of a diagram consists of

- (`lift`) For any cone $c$, a map `lift` : $c \to l$ in the category $\mathcal{C}$.
- (`fac`) A proof that `lift` commutes with everything else in the diagram.
- (`uniq`) Any other map $c \to l$ commuting with everything is equal to `lift`.

Put another way, a limit of a diagram is the terminal object when considering the cones as a category.

```
structure is_limit (t : cone F) :=
(lift  : Π (s : cone F), s.X ⟶ t.X)
(fac'  : ∀ (s : cone F) (j : J), lift s ≫ t.π.app j = s.π.app j . obviously)
(uniq' : ∀ (s : cone F) (m : s.X ⟶ t.X) (w : ∀ j : J, m ≫ t.π.app j = s.π.app j),
m = lift s . obviously)
```

Note that `lift` is data, whereas the others are propositional. Also note that `fac'` and `uniq'` are appended with `obviously`. This means that when we construct a limit, we could only supply the `lift`, and let `lean` figure the rest out. Many of the structures in the category theory library use this so that making instances of categories, functors, natural transformations, and limits less work.

$$\begin{tikzcd}
 & c \arrow[d, dashed] \arrow[ddl] \arrow[ddr] & \\
 & l \arrow[dl] \arrow[dr] & \\
\mathcal{F}i \arrow[rr, bend left, "\mathcal{F}\psi"] \arrow[rr, bend right, "\mathcal{F}\phi"'] & & \mathcal{F}j
\end{tikzcd}$$

$l$ is the equalizer of $\mathcal{F}\phi$ and $\mathcal{F}\psi$