Type Inhabitation as Existence of Normal Identity Paths

by Sven Nilsen, 2019

In this paper I show that type inhabitation can be interpreted as the existence of normal identity paths.

The `∃?` operator returns `true` if a normal path[2] exists and `false` otherwise[1]:

$$\exists?f[g_{i \to n}] : bool$$

A symmetric path[2] of `f` by `id` is the same as `f`:

$$f[id] <=> f$$

Therefore, the `∃?` operator can be interpreted as the existence of the normal identity path:

$$\exists?f[id] <=> \exists?f$$

The normal identity path exists if and only if the type of `f` is inhabited[3].

This works also when `f` is a constant of some type `T`:

$$f[id] <=> f[id \to id] <=> f[unit \to id]$$

$\exists?f[id] <=> \exists?f[unit \to id]$ Existence of normal identity path of a constant is type inhabitation

$f : () \to T$ Constants can be thought of functions with zero arguments

When `f` is a constant, the `id` applied to the arguments `()` returns `()`, which is same as `unit`[4].
With other words, the arguments are erased while the output is not,
so the normal path exists if and only if the value of `f` inhabits the type.

This means that the `∃?` operator is the same as checking for type inhabitation in general.

References:

[1]   "Existence of Normal Paths"
        Sven Nilsen, 2019
        https://github.com/advancedresearch/path_semantics/blob/master/papers-wip/existence-of-normal-paths.pdf

[2]   "Normal Paths"
        Sven Nilsen, 2019
        https://github.com/advancedresearch/path_semantics/blob/master/papers-wip/normal-paths.pdf

[3]   "Type inhabitation"
        Wikipedia
        https://en.wikipedia.org/wiki/Type_inhabitation

[4]   "The Terminal Function & Propositions of Irrelevance"
        Sven Nilsen, 2017-2019
        https://github.com/advancedresearch/path_semantics/blob/master/papers-wip/the-terminal-function-and-propositions-of-irrelevance.pdf