

Catuṣkoṭi Existential Path Equations

by Sven Nilsen, 2021

In this paper I introduce `both` and `neither` for many-value logic in existential path equations.

An existential path equation^[1] with an indetermined function identity `f`:

$$f \Rightarrow g$$

Is a shorthand syntax equivalent to the following:

$$\exists f \Leftrightarrow g$$

On the other hand, when `g` is a constant, e.g. `true`, one says:

$$\exists f \Leftrightarrow (= \text{true})$$

Which means that the identity of `f` is determined:

$$f \Leftrightarrow \text{true}$$

This is used as a shorthand syntax to say that “`f` returns `true` for all inputs”.

In Catuṣkoṭi^[2], it is useful to express that “`f` returns `both`” or “`f` returns `neither`”:

| | |
|---|--|
| $f \Rightarrow \text{both}$ | $f \Rightarrow \text{neither}$ |
| $\exists f \Leftrightarrow \text{true}$ | $\exists f \Leftrightarrow \text{false}$ |
| $f \Rightarrow \text{true}$ | $f \Rightarrow \text{false}$ |

It follows that `both` and `neither` might be defined as following:

$$\begin{aligned} \text{both} &:= \text{true} \\ \text{neither} &:= \text{false} \end{aligned}$$

The function `neither` is determined from the existential path.

The function `both` is undetermined from the existential path for inhabited types.

In ternary many-value logic with `true`, `false`, `unknown`, the `unknown` is usually defined as `both`:

$$\text{unknown} := \text{true} \qquad \text{unknown} \Leftrightarrow \text{both}$$

This corresponds to Kleene’s “strong logic of indeterminacy”.

In Priest’s “logic of paradox”, `unknown` is defined as `neither`.^[3]

References:

- [1] “Higher Order Operator Overloading and Existential Path Equations”
Sven Nilsen, 2019
https://github.com/advancedresearch/path_semantics/blob/master/papers-wip/higher-order-operator-overloading-and-existential-path-equations.pdf

- [2] “Catuskoṭi”
Wikipedia
<https://en.wikipedia.org/wiki/Catu%E1%B9%A3ko%E1%B9%ADi>

- [3] “Kleene and Priest logics – Three-value logic”
Wikipedia
https://en.wikipedia.org/wiki/Three-valued_logic#Kleene_and_Priest_logics