Path Function Existence

by Sven Nilsen, 2020

In this paper I prove that path sub-types form the existence of a function.

Assume the following function with path sub-types[1]:

f(a : [g] c) → [g] d(c) { b }

If `d` is a valid function of `c`, `f[g] <=> d`. One can model the existence in Path Semantical Logic[2]:

∴ (f, g, a_0, a_1, b_0, b_1, c_0, c_1, d_0, d_1) (A, B, C, D):
fun(f, a_0, A, b_0, B),
fun(f, a_1, A, b_1, B),
f=>(a_0 ⩡ a_1),
fun(f ∧ g, a_0, A, c_0, C),
fun(f ∧ g, a_1, A, c_1, C),
fun(f ∧ g, b_0, B, d_0, D),
fun(f ∧ g, b_1, B, d_1, D),
(f, g)=>(c_0 ⩡ c_1)
---
fun(f ∧ g, c_0, C, d_0, D),
fun(f ∧ g, c_1, C, d_1, D)

∵ fun(f, a, A, b, B) = c=>(a=>b, a(A)=>b(B))

Here, variables starting with a small letter is level 1, and variables starting with a big letter is level 0.
The notation `a(A)` means `a=>A` where `A` is at a lower level.
Comma outside calling arguments when calling functions is the same as `∧` (Logical AND).
For modeling functions in Path Semantical Logic, see paper "Modeling Functions"[3].

This is not provable in normal Propositional Logic[4].
However, it is an almost-tautology[5], with only 2 out of 16 384 (2^{14}) cases being `false`.

When substituting e.g. `$c_1$` with `$c_0$`, the function `d` is no longer valid,
because `$d_0$` and `$d_1$` are exclusive to each other.
For any function `f`, if `x = y`, then `f(x) = f(y)`.
Path Semantical Logic models also this case correctly.

When substituting `$c_1$` with `$c_0$` and `$d_1$` with `$d_0$`, the function `d` is valid again.
This is also modelled correctly in Path Semantical Logic.

References:

[1] “Sub-Types as Contextual Notation”
Sven Nilsen, 2018
https://github.com/advancedresearch/path_semantics/blob/master/papers-wip/sub-types-as-contextual-notation.pdf

[2] “Path Semantical Logic”
AdvancedResearch, reading sequence on Path Semantics
https://github.com/advancedresearch/path_semantics/blob/master/sequences.md#path-semantical-logic

[3] “Modeling Functions”
Sven Nilsen, 2020
https://github.com/advancedresearch/path_semantics/blob/master/papers-wip/modeling-functions.pdf

[4] “Propositional calculus”
Wikipedia
https://en.wikipedia.org/wiki/Propositional_calculus

[5] “Tautology (logic)”
Wikipedia
https://en.wikipedia.org/wiki/Tautology_(logic)