Noncontractible Types

by Sven Nilsen, 2020

In this paper I show that there is a way of modeling noncontractible types in Path Semantical Logic.

For example, the following is a proof in Path Semantical Logic[1]:

$$(a, c)\ (A, B, C):$$
$$a(A),\ c(C),\ a=c,\ \mathbf{\neg(A \land B \land C) \land (A \lor B \lor C)}\ \Rightarrow\ (A \veebar B) \land (B \veebar C)$$

Here, the tuple `(a, c)` has level 1 and the tuple `(A, B, C)` has level 0.
The notation `a(A)` means `a=>A` where `A` is at a lower level.

The expression `**¬(A ∧ B ∧ C) ∧ (A ∨ B ∨ C)**` models a noncontractible type family `A, B, C`.
It means that there are at least two types among `A, B, C` that can not be proven equal.
Since `a=c`, `A=C`, therefore `(A ⊻ B) ∧ (B ⊻ C)` since `B` is not equal to both `A` and `C`.

In the paper "Proof of Exclusiveness"[2], I showed that NAND models proof of noncontractibility.

A family of types are noncontractible when, in any context, there exists some membership in at least one type, that does not exist in some other type. This means that the types can not be proven equal.

According to the core axiom of Path Semantics[3], all undefined symbols have same meaning[4].
As a consequence, to model noncontractibility correctly, one must exclude the undefined case.

The relation of noncontractiblity for two propositions `A` and `B` can be expressed as following:

$$\neg(A \land B) \land (A \lor B) \qquad \Leftrightarrow \qquad A \veebar B$$

For more than two arguments, XOR is too strong and one must use the following:

$$\neg(X_0 \land X_1 \land \ldots \land X_{n-1}) \land (X_0 \lor X_1 \lor \ldots \lor X_{n-1})$$

Written with first-order logic[5] syntax:

$$\neg\forall\ i\ \{\ X_i\ \} \land \exists\ i\ \{\ X_i\ \}$$

For example, if `A, B` are noncontractible, then it implies that `A, B, C` are noncontractible.
The family `A, B, C` can never be proven equal to each other since `A, B` can never be proven equal.

However, if `A, B, C` are noncontractible, then this does not imply that `A, B` are noncontractible.
On the other hand, if `A, B` are equal and `A, B, C` are noncontractible, then `A, C` and `B, C` are noncontractible.

References:

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

[2] "Proof of Exclusiveness"
Sven Nilsen, 2017
https://github.com/advancedresearch/path_semantics/blob/master/papers-wip/proof-of-exclusiveness.pdf

[3] "Path Semantics"
Sven Nilsen, 2016-2019
https://github.com/advancedresearch/path_semantics/blob/master/papers-wip/path-semantics.pdf

[4] "Undefined Symbols"
Sven Nilsen, 2019
https://github.com/advancedresearch/path_semantics/blob/master/papers-wip/undefined-symbols.pdf

[5] "First-order logic"
Wikipedia
https://en.wikipedia.org/wiki/First-order_logic