Concrete and Abstract Transport

by Sven Nilsen, 2020

In this paper I show that there 8 concrete and 4 abstract binary relations in Path Semantical Logic.

In Path Semantical Logic[1], there are two kinds of transports:

- Concrete transport using the analogue of (Normal) Implication Theorem[2]
- Abstract transport using the analogue of Abstract Implication Theorem[3]

There are 8 binary relations that transports concretely:

and, fst, snd, or, eq, rimply, imply, true_2

There are 4 binary relations that transports abstractly:

eq, rimply, imply, true_2

The `$\text{false}_2$` relation is special, because it allows one to prove anything, including `$\text{false}_2$`.

Here is a table overview of all 16 functions of type `$\mathbb{B}^2 \to \mathbb{B}$`:

Name/Expression	**Binary code**	**Concrete**	**Abstract**
false_2	0000	yes	yes
and	0001	yes	no
exc	0010	no	no
fst	0011	yes	no
not . rimply	0100	no	no
snd	0101	yes	no
xor	0110	no	no
or	0111	yes	no
nor	1000	no	no
eq	1001	yes	yes
not . snd	1010	no	no
rimply	1011	yes	yes
not . fst	1100	no	no
imply	1101	yes	yes
true_2	1111	yes	yes

References:

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

[2] “Implication Theorem”
Sven Nilsen, 2020
https://github.com/advancedresearch/path_semantics/blob/master/papers-wip/implication-theorem.pdf

[3] “Abstract Implication Theorem”
Sven Nilsen, 2020
https://github.com/advancedresearch/path_semantics/blob/master/papers-wip/abstract-implication-theorem.pdf