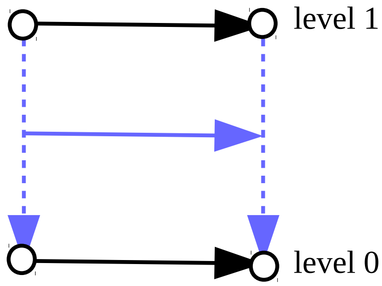
Visualizing Implication Theorems

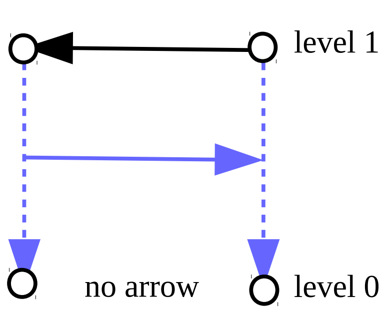
by Sven Nilsen, 2020

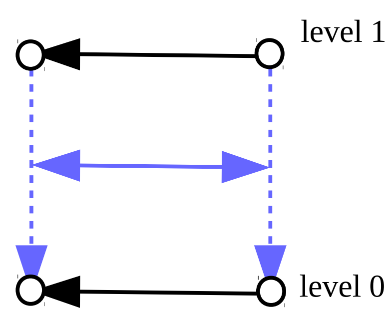
Path Semantical Logic[1] contains 3 implication theorems, of which the most general one is the Constrained Implication Theorem[2]. A way to visualize how it works is the following:



Each arrow represents a material implication, e.g. `imply(a, b)` or `a=>b`.
A continuous arrow is asserted directly, e.g. `**a=>b**`.
A dashed arrow is only used indirectly, by being a sub-expression, e.g. `(**a=>b**)=>(c=>d).

The blue arrows defines a "surface" between level 1 and level 0. This surface behaves like a filter. It allows relations in level 1 to propagate down to level 0. If the arrow at level 1 points in opposite direction, then the relation gets blocked, but if one uses a bidirectional arrow[3] (an equality), any sort of arrow can pass through the filter:





In Path Semantical Logic, it is sufficient to construct the surface to allow relations to propagate through. The truth value of propositions at the nodes does not need to pass through, but you can also do that, by turning the dashed vertical arrows into continuous ones[4].

In normal Propositional Logic[5], one is used to think about truth values as being the "cause" of propagation, e.g. in modus ponens, `P=>Q, P  =>  Q`. Here, `Q` is true because `P` is true and there is a law `P=>Q` that allows truth of `P` to propagate to the truth of `Q`. However, this method allows only transport of truth values. To do the same for directional relations would require 3 laws instead of one, since there are 3 such relations: `A=>B`, `B=>A` and `A=B`. For example, `**(A=>B)**=>(Q=>R)` for the case `**A=>B**`. In Path Semantical Logic, the rule `(A=>Q)=(B=>R)` handles all 3 cases.

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

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

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

[5] “Propositional Calculus”
Wikipedia
https://en.wikipedia.org/wiki/Propositional_calculus