5

15th Jan'24

BHK Interpretation of Minimal Logic

Constructions proved by BHK often use some fun^n as input and output some other fn.

$$(A \to B) \to (\neg B \to \neg A)$$

These are called higher order fn.

Ordinary math does not have any notation to deal with such a fn

↳ There is a convenient notation λ notation which we can use

<u>Example</u>

$x+y$ depends on x, y. free variables of this expression.

$\lambda x . x+y$ Free variable y

(x is no longer a free var.)
closed terms

specifies for each y_0, a fn of one argument.

$x \mapsto x + y_0$

$A \vdash (B \to A)$

$$\dfrac{\dfrac{x{:}A \vdash A{:}x}{x{:}A, \neg B \vdash A{:}x}}{A \vdash (B \to A)}$$

$\lambda y . \lambda x . x + y$

No free variables
(all closed terms)

This is a fu^n which takes y as an argument, and gives out fn^n, $\lambda x . x + y_0$ as output

All output fu^n are fn^{n}s of <u>one</u> argument

$\lambda y : \mathbb{R}\ \lambda x : \mathbb{R} .\ x + y$

Eg:

$A \to (B \to A)$ construction for this is,

$\lambda x : A\ \lambda y : B . x\ :\ A \to (B \to A)$

Eg:

$A \to \neg\neg A$

$\underbrace{\lambda x : A\ \underbrace{\lambda y : A \to \bot\ .\ yx}_{\neg A \to \bot}}_{A \to (\neg A \to \bot)}$

yx is application of y to x.

(Instead of $y(x)$, we just write x.)

fu^n application associates to the left.

Eg:

$(A \to B) \to (\neg B \to \neg A)$

$\lambda x : A \to B\ \lambda y : \neg B\ \lambda z : A\quad y(xz)$

$$
\cfrac{\cfrac{\cfrac{\cfrac{x : A \to B \vdash x : A \to B \quad , \quad y : A \vdash y : A}{x : A \to B,\ y : A \vdash xy : B} \quad , \quad z : \neg B \vdash z : \neg B}{x : A \to B,\ y : A,\ z : \neg B \vdash \perp z(xy)}}{x : A \to B,\ z : \neg B \vdash \neg A \quad \lambda y : A.\ z(xy)}}{A \to B \vdash \neg B \to \neg A}
$$

functional languages : Haskell

Logic	PL
Proposition	Types
Construction	Program
Argument deri= of proposition	Type assig= to program
Normalization	Evaluation

↳ Curry-Howard Isomerisation

Theorem

Every proposition derivable in IL, admits a BHK interpretation

Proof:

By induction on size of derivation of the propositions.

Base case: $x : A \vdash x : A$

Induction step: case depending on the last rule in the derivation.

$$\frac{\Gamma \vdash A \qquad \Gamma' \vdash B}{\Gamma, \Gamma' \vdash A \wedge B}.$$

$$\frac{\Gamma \vdash A \to B \;,\; \Gamma' \vdash A}{\Gamma, \Gamma' \vdash B}.$$

Exercise

1. Complete the other cases of Induction step
2. Also state the theorem more precisely and one complete proof.