Reflection and its use

from science to meditation

Review

Music

Pierluigi da Palestrina (1525?-1594)

Ave Verum

Traditional Buddhist Music (Shakuhachi, Japan)

Igor Stravinsky

Introitus (1965) In memoriam for T.S. Eliot

Contents

Propositional Logic

Introduction Rules		Elimination Rules
$\dfrac{\Gamma, A \vdash B}{\Gamma \vdash (A \to B)}$		$\dfrac{\Gamma \vdash (A \to B) \quad \Gamma \vdash A}{\Gamma \vdash B}$
$\dfrac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash (A \,\&\, B)}$		$\dfrac{\Gamma \vdash (A \,\&\, B)}{\Gamma \vdash A}$ $\dfrac{\Gamma \vdash (A \,\&\, B)}{\Gamma \vdash B}$
$\dfrac{\Gamma \vdash A}{\Gamma \vdash (A \vee B)}$	$\dfrac{\Gamma \vdash B}{\Gamma \vdash (A \vee B)}$	$\dfrac{\Gamma \vdash (A \vee B) \quad \Gamma, A \vdash C \quad \Gamma, B \vdash C}{C}$
Start Rule	**Absurdum Rule**	**Classical Negation**
$\dfrac{A \in \Gamma}{\Gamma \vdash A}$	$\dfrac{\Gamma \vdash \bot}{\Gamma \vdash A}$	$\dfrac{\Gamma, \neg A \vdash \bot}{\Gamma \vdash A}$ $\quad \neg A := (A \to \bot)$

Example

$$A \to (A \to B) \vdash A \to B$$

Predicate Logic

This is all of logic (Aristotle, Boole, Peirce, Frege, Brouwer, Heyting, Gentzen)

	Introduction Rules		Elimination Rules
$\rightarrow$	$\dfrac{\Gamma, A \vdash B}{\Gamma \vdash (A \rightarrow B)}$		$\dfrac{\Gamma \vdash (A \rightarrow B) \quad \Gamma \vdash A}{\Gamma \vdash B}$
$\&$	$\dfrac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash (A \ \& \ B)}$		$\dfrac{\Gamma \vdash (A \ \& \ B)}{\Gamma \vdash A}$ $\quad$ $\dfrac{\Gamma \vdash (A \ \& \ B)}{\Gamma \vdash B}$
$\vee$	$\dfrac{\Gamma \vdash A}{\Gamma \vdash (A \vee B)}$	$\dfrac{\Gamma \vdash B}{\Gamma \vdash (A \vee B)}$	$\dfrac{\Gamma \vdash (A \vee B) \quad \Gamma, A \vdash C \quad \Gamma, B \vdash C}{C}$
$\forall$	$\dfrac{\Gamma \vdash A}{\Gamma \vdash \forall x.A}$ $x \notin \Gamma$		$\dfrac{\Gamma \vdash \forall x.A}{\Gamma \vdash A[x := t]}$ t is free in A
$\exists$	$\dfrac{\Gamma \vdash A[x := t]}{\Gamma \vdash \exists x.A}$		$\dfrac{\Gamma \vdash \exists x.A \quad \Gamma, A \vdash B}{\Gamma \vdash B}$ $x \notin B$
	Start Rule	Absurdum Rule	Classical Negation
	$\dfrac{A \in \Gamma}{\Gamma \vdash A}$	$\dfrac{\Gamma \vdash \bot}{\Gamma \vdash A}$	$\dfrac{\Gamma, \neg A \vdash \bot}{\Gamma \vdash A}$ $\neg A := (A \rightarrow \bot)$

Example

$$A \to (A \to B) \vdash A \to B$$

Can be applied in connection with

$$\begin{array}{rcll}
\forall x, y.(x > y) & \to & \neg(y > x) & \quad\text{axiom} \\
\forall x, y.x > y & \to & (y > x \to \bot) & \\
x > x & \to & (x > x \to \bot) & \\
x > x & \to & \bot & \\
\neg(x > x) & & & \\
\forall x.\neg(x > x) & & & \quad\text{consequence}
\end{array}$$

The Turing Test

"Does a machine have consciousness?"

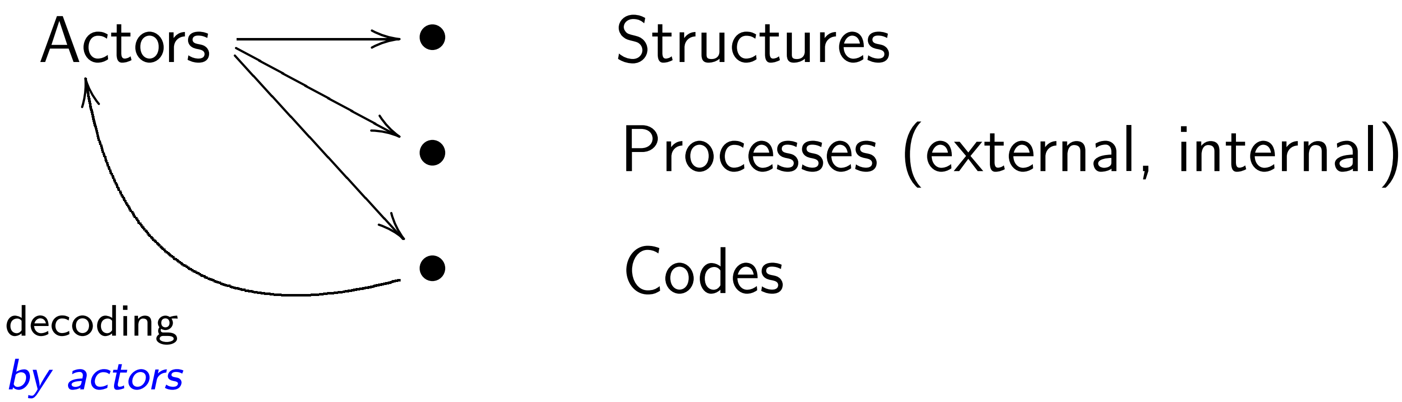
In order to make this question more precise
Turing modified the *simulation test*:
find out while chatting whether two persons are male or female,
while one is simulating to be the other gender

Turing's version: decide between a human and a robot
if we cannot distinguish correctly, we may say that
the robot has consciousness

Sharp reflection

Reflection

Global feedback



Ordinarily it does not always happen that actors modify the code

In *sharp* reflection it does

Sharp reflection in language

Rebus (in Dutch)

		$\longmapsto$	stoomboot
b=st	m=t		

The following correction process can be captured into words

"Hij bedoeld iets" $\longmapsto$

"Hij bedoel~~d~~ iets" X t $\longmapsto$

"Hij bedoelt iets"

Sharp reflection in Computing

A *compiler* takes a program

and transforms it into another one

directly interpretable by the electronics on a chip

Reverse reflection

actor $\longmapsto$ code

Reverse reflection in mathematics

Continuous functions (can write graph whithout lifting pen from paper)

$\sin(x)$
$\sin(x) + \sin(2x)$
$\sin(x) + \sin(2x) + \sin(4x)$
$\ldots$

Theorem

$$\forall e \in L. [\![e]\!] \text{ is continuous in } x$$

Here L is the language consisting of the names of such functions:

$$\{\sin, \sin_2, \sin_3, \ldots\}$$

closed under arbitrary sums

In order to show that for example $\sin(x) + \sin(4x)$ is continuous we write it as

$$\sin(x) + \sin(4x) = [\![\sin + \sin_4]\!]$$

and apply the theorem. We need to go from the actor to its code!

The quote mode

To go from an actor to its code is possible, if we have that code already in one way or another. We may suppose that every mathematical object (actor) is given as its code and is interpreted whenever convenient or necessary.

This actually happens in a version of the programming language LISP and is called the quote-mode of this language.

Increased states of mindfulness also work like this.