Modes of Truth, Ways of Knowing

Giuseppe Primiero

FWO - Flemish Research Foundation
Centre for Logic and Philosophy of Science, Ghent University
IEG - Oxford University

Giuseppe.Primiero@Ugent.be
http://www.philosophy.ugent.be/giuseppeprimiero/

ENS, Paris, France
23 November, 2011

Modes of Truth

The theory of truth-makers is a general realist model of *Moments* as neutral bearers of truth, based on the Aristotelian distinction from the *Categories* between substiantial and accidental entities [Mulligan et al., 1984]:

Modes of Truth

The theory of truth-makers is a general realist model of *Moments* as neutral bearers of truth, based on the Aristotelian distinction from the *Categories* between substiantial and accidental entities [Mulligan et al., 1984]:

> *"a is a moment iff a exists and a is de re necessarily such that either it does not exist or there exists at least one object b, which is de re possibly such that it does not exist and which is not a proper or improper part of a. In such a case, b is a fundament of a, and we say also that b founds a or a is founded on b. If c is any object containing a fundament of a as proper or improper part, but not containing a as proper or improper part, we say, following Husserl, that a is dependent on c. Moments are thus by definition dependent on their fundaments. Objects which are not moments we call independent objects or substances"*

Modes of Truth (3)

In the tradition that stems (at least) from *De Interpretatione*, a natural expansion is considered towards various sentence-type: modal, temporal, counterfactual, intentional, deictic assertions.

Modes of Truth (3)

In the tradition that stems (at least) from *De Interpretatione*, a natural expansion is considered towards various sentence-type: modal, temporal, counterfactual, intentional, deictic assertions.

Questions:

1. Do moments presuppose or assume other moments? How are these logical relations expressed?
2. how do sentences express modes of moments?
3. A cube being white corresponds to the whiteness of the cube and two objects colliding is an equivalent moment to their collision; what about the state or moment that corresponds to the possibility of collision and the necessity of being white? Do these expressions have corresponding moments?

Modes of Truth (4)

A theory of ‘modes of truth’ can be formulated in one of the many classical logic formats that allow for modalities; but I will not try to illustrate one and use the previous questions only as a suggestion to move to other questions:

Modes of Truth (4)

A theory of ‘modes of truth’ can be formulated in one of the many classical logic formats that allow for modalities; but I will not try to illustrate one and use the previous questions only as a suggestion to move to other questions:

Questions:

1. Are antirealist approaches to truth – and in particular theories of proof-objects – truly unfit for developing theories of empirical truth?
2. Do we really want one such theory?

Ways of Knowing (1)

I shall argue that the epistemic articulation of an intuitionistic theory of proof-objects is deep enough to account for various ways in which knowledge is obtained. In particular, I shall maintain that:

Ways of Knowing (1)

I shall argue that the epistemic articulation of an intuitionistic theory of proof-objects is deep enough to account for various ways in which knowledge is obtained. In particular, I shall maintain that:

1. the notion of proof-object is variegated enough to account for qualitatively distinct epistemic attitudes;
2. weaker states as the one of 'admissible knowledge' can be formulated;
3. local and contextual validity can be defined as to express limited knowability;
4. finally, the previous points ground a theory of epistemic states fit for empirical knowledge.

Outline

1. Conditions for Knowing
2. When Conditions are (and are not) satisfiable
3. A refiniment of constructive epistemology

Actuality and Potentiality

A constructive theory of proof-objects endorses a dynamic epistemology by admitting the process of constructing as crucial to its underlying theory of truth-bearers:

Actuality and Potentiality

A constructive theory of proof-objects endorses a dynamic epistemology by admitting the process of constructing as crucial to its underlying theory of truth-bearers:

> *"[. . .] there is an absolutely clear order of conceptual priority between these two notions of [1 actual and 2 potential] existence [. . .] in that of course the notion of existence in sense 1 is presupposed in 2, because to say that a exists actually is [. . .] the same as to say that this judgement is known and hence that a exists in sense 1 is contained as a component in a exists actually, and on the other hand there is a similar phenomenon [. . .] because to say that a exists potentially is to say that the judgement a [a] exists can be known and when you say that it can be known that means of course that it can be know actually, it can actually be known [. . .] and hence the notion of actual being or actual existence is prior conceptually to the notion of potential existence". [Martin-Löf, 1993]*

Actual and Potential formulation of proof-objects

Though there is no strict sense in which a proof-object is potential, the following abstraction process is validly formulated:

- A proof-object testifies for the (actual) truth of a certain propositional content;
- potential truth corresponds to the *potential formulation* of a proof-object;

Actual and Potential formulation of proof-objects

Though there is no strict sense in which a proof-object is potential, the following abstraction process is validly formulated:

- A proof-object testifies for the (actual) truth of a certain propositional content;
- potential truth corresponds to the *potential formulation* of a proof-object;
  - forgetting the computational content of a proof-object, one can *assume* to know it;
  - Using such an assumption, presupposes its computational content to be meaningful (constructible).

Stages of Assertions

Hence knowledge can be articulated in the following stages:

1. the assertion of the existence of a certain proof-object;
2. the assertion of an assumption on the existence of a certain proof-object;
3. the assertion of an assumption on the knowledge of a closed derivation for a certain proof-object;
4. the assertion of a presupposition needed by the the existence of a certain proof-object.

Articulating Ways of Knowing

The first articulation of ways of knowing is therefore based on the (mostly well-known) theory of conditions for knowledge in terms of proof-obejcts, by way of the following notions ([Primiero, 2004]):

1. alethic assumptions;
2. epistemic assumptions;
3. presuppositions of meaning.

Articulating Ways of Knowing

The first articulation of ways of knowing is therefore based on the (mostly well-known) theory of conditions for knowledge in terms of proof-obejcts, by way of the following notions ([Primiero, 2004]):

1. alethic assumptions;
2. epistemic assumptions;
3. presuppositions of meaning.

Both the notion of assumption and the analysis of conditions for knowledge lead us to the crucial issue of hypothetical judgement (see [van Atten, pear]; [Primiero, 2009b])

Hypothetical Reasoning

The notions of dependent type and dependent object are standardly introduced in CTT to analyze hypothetical judgments. Let us recall that a dependent judgment is introduced in CTT as an expression of the following form:

$$\beta \; \textit{true}[x_1 : \alpha_1, x_2 : \alpha_2, \ldots, x_n : \alpha_n]$$

This reflects the structure of a consequence, as the holding of the truth of the conclusion given the truth of the antecedents:

$$A_1 \; \textit{true}, \ldots, A_n \; \textit{true} \Rightarrow B \; \textit{true}.$$

Alethic vs. Epistemic Assumptions [Primiero, 2004]

- ‘Assume to know a proof of *A*’ has an epistemic value; this is very often conflated with the notion of something needed to be known for something else to be known (see [Sundholm, 2004]);

Alethic vs. Epistemic Assumptions [Primiero, 2004]

- 'Assume to know a proof of *A*' has an epistemic value; this is very often conflated with the notion of something needed to be known for something else to be known (see [Sundholm, 2004]);
- Epistemic Assumptions refer to something *really true*, an assumption of a knowable judgment, or the assumption about possessing knowledge of the proof object for the related content; in natural deduction these expressions are equivalent to implications presenting closed derivations for the antecedent;

Alethic vs. Epistemic Assumptions [Primiero, 2004]

- ‘Assume to know a proof of *A*’ has an epistemic value; this is very often conflated with the notion of something needed to be known for something else to be known (see [Sundholm, 2004]);
- Epistemic Assumptions refer to something *really true*, an assumption of a knowable judgment, or the assumption about possessing knowledge of the proof object for the related content; in natural deduction these expressions are equivalent to implications presenting closed derivations for the antecedent;
- one is relying on the actual proof for the proposition used as antecedent.

Alethic vs. Epistemic Assumptions [Primiero, 2004]

- 'Assume A to be true' reflects the usual understanding of derivations in natural deductions: demonstrating a certain implication $A \supset B$ starting from the antecedent that A is true does not exclude that the set of proofs for A may be *actually* empty (Cf. [Sundholm, 2004, p.451]);

Alethic vs. Epistemic Assumptions [Primiero, 2004]

- 'Assume A to be true' reflects the usual understanding of derivations in natural deductions: demonstrating a certain implication $A \supset B$ starting from the antecedent that A is true does not exclude that the set of proofs for A may be *actually* empty (Cf. [Sundholm, 2004, p.451]);
- Alethic assumptions are weaker

Alethic vs. Epistemic Assumptions [Primiero, 2004]

- ‘Assume A to be true’ reflects the usual understanding of derivations in natural deductions: demonstrating a certain implication $A \supset B$ starting from the antecedent that A is true does not exclude that the set of proofs for A may be *actually* empty (Cf. [Sundholm, 2004, p.451]);
- Alethic assumptions are weaker
- an alethic assumption $[x : A]$ does not necessarily involve the necessary existence of the related proof object.

Conceptual Priority: Presuppositions [Primiero, 2004]

To proceed in stating something to be known, we need to establish its predicability.

Conceptual Priority: Presuppositions [Primiero, 2004]

To proceed in stating something to be known, we need to establish its predicability.

$$[x_1 : \alpha_1]$$
$$\alpha_2 : \mathit{type}$$

states that α_2 is a type depending on the assumption that a certain object a_1 substituted for x_1 belongs to the type α_1. This assumption is itself based on a *presupposition*, namely the judgment

$$< \alpha_1 : \mathit{type} >$$

which states predicability of type α_1. Hence the condition that a term a_1 can be found.

Conceptual Priority: Presuppositions [Primiero, 2004]

To proceed in stating something to be known, we need to establish its predicability.

$$[x_1 : \alpha_1]$$
$$\alpha_2 : \textit{type}$$

states that α_2 is a type depending on the assumption that a certain object a_1 substituted for x_1 belongs to the type α_1. This assumption is itself based on a *presupposition*, namely the judgment

$$< \alpha_1 : \textit{type} >$$

which states predicability of type α_1. Hence the condition that a term a_1 can be found.

Definition

Presuppositions state that the types involved are apt to be predicated, where predication aptness indicates being at disposal for (right or wrong) predication

Conceptual Priority: Presuppositions

Judgments	Immediate Presuppositions
$\emptyset$	none
$x : \alpha$	$< \alpha : type >$
$[\Gamma]$ $\alpha : type$	$< \Gamma : context >$
$[\Gamma]$ $\alpha = \beta : type$	$[\Gamma], [\Gamma]$ $< \alpha : type >, < \beta : type >$
$[\Gamma]$ $a : \alpha$	$[\Gamma]$ $< \alpha : type >$
$[\Gamma]$ $a = b : \alpha$	$[\Gamma], [\Gamma]$ $< a : \alpha >, < b : \alpha >.$

Presupposition as an epistemic notion

Definition

The judgment $< J_1 >$ is a presupposition for the judgment J_2 if the assertion condition of J_2 depends on J_1's being known and so J_2 is a judgment-candidate once J_1 is known.

Presupposition as an epistemic notion

Definition

The judgment $< J_1 >$ is a presupposition for the judgment J_2 if the assertion condition of J_2 depends on J_1's being known and so J_2 is a judgment-candidate once J_1 is known.

A judgment is a candidate if its presuppositions are known; alternatively, aptness for predication in a judgment ($a : \alpha$) expresses meaningfulness of a certain type (α : *type*).

Conditions and Attitudes

1. the notion of proof-object is a stratified one;
2. verification is not a condition expressed by a stand-alone object, rather it is a process which requires to lay down different conditions for the propositional content at hand;
3. conditions formalize epistemic attitudes:
   - provability has its natural counterpart in an epistemic modality of necessity
   - *knowledge formulated under assumptions* can be expressed by means of a judgemental modality of possibility ([Primiero, 2009b])

Conditional Knowing and Possibility

A coherent weakening of the constructive basis for defining truth is admissible by referring to partial termination for proof-objects. A proof-object *a* for *A* is said to partially terminate (in this sense) if the conditions Γ that need to be satisfied in order the truth of *A* to be asserted

1. are established,
2. it is known that such conditions can be satisfied,
3. it is not known yet if they are actually satisfied (hence not actually known).

Conditional Knowing and Possibility

A coherent weakening of the constructive basis for defining truth is admissible by referring to partial termination for proof-objects. A proof-object a for A is said to partially terminate (in this sense) if the conditions Γ that need to be satisfied in order the truth of A to be asserted

1. are established,
2. it is known that such conditions can be satisfied,
3. it is not known yet if they are actually satisfied (hence not actually known).

Conditions are accepted and taken for valid, *unless proven otherwise*: the corresponding judgement is of the form

Possibly, A is true.

Refutable Assumptions [Primiero, 2012]

Hence we obtain two forms of validity w.r.t. assumptions:

Refutable Assumptions [Primiero, 2012]

Hence we obtain two forms of validity w.r.t. assumptions:

- a judgement $\Box(A\ true)$ expresses that a content *A* is true in any epistemic state, as *A* is independent from any refutable condition (either there are none, or all of them have been secured):

  (I know that) S is P, given that I know that A_1 *to* A_n*;*

Refutable Assumptions [Primiero, 2012]

Hence we obtain two forms of validity w.r.t. assumptions:

- a judgement $\Box(A\ true)$ expresses that a content A is true in any epistemic state, as A is independent from any refutable condition (either there are none, or all of them have been secured):

  (I know that) S is P, given that I know that A_1 to A_n;

- a judgement $\Diamond(A\ true)$ expresses that a content A is true in some epistemic states, namely where certain conditions are not refuted:

  (I know that) S is P, provided that A_1 to A_n are not refuted.

Refutable Assumptions [Primiero, 2012]

Definition (Computational Rules for Modal Judgements)

$$\frac{}{\Gamma, a : A, \Delta \vdash A\ true} \text{ Premise Rule}$$

$$\frac{}{\Gamma, x : A, \Delta \vdash A\ true^*} \text{ Hypothesis Rule}$$

$$\frac{a : A}{\Box(A\ true)} \ \Box\text{-Formation} \qquad \frac{x : A}{\Diamond(A\ true)} \ \Diamond\text{-Formation}$$

$$\frac{\Gamma \vdash A\ true}{\Box\Gamma \vdash \Box(A\ true)} \ \mathrm{I}\Box \qquad \frac{\Box\Gamma \vdash \Box(A\ true) \quad \Delta, a : A \vdash B\ true}{\Gamma, \Delta \vdash B\ true} \ \mathrm{E}\Box$$

$$\frac{\Gamma, x : A \vdash B\ true^*}{\Box\Gamma, \Diamond(A\ true) \vdash \Diamond(B\ true)} \ \mathrm{I}\Diamond$$

$$\frac{\Box\Gamma, \Diamond\Delta \vdash \Diamond(A\ true) \quad \Delta, x : A \vdash B\ true^*}{\Gamma, \Delta \vdash B\ true^*} \ \mathrm{E}\Diamond$$

Localized Validity

- further refinement: determing the scope of validity of conditons;
- state explicitely which further extensions of a context Γ for a judgement A *true* can be given such that A *true* is still valid;
- express local validity of computational processes, by adding indexing on terms and in turn on modalities to express agents or locations;
- by this latter task, one induces explicitely aspects of failure and interaction, by referring to a complete mapping of the levels of validity admitted by a judgement, [Primiero, 2013].

Modalities for localized computations [Primiero, 2011]

- Procedural Semantics with Modalities for Contextual (localized) Computing;
- designed from a multi-modal type system with a BHK semantics Martin-Löf's style with Proofs-as-Programs;
- localization of processes to represent distributed computing;
- rules for connectives intepret composition of processes;
- modal rules interpret interaction of code at locations (mobility).

Other Extended Semantics

- (Modal Types based) Dynamic Semantics in terms of a big-step evaluation relation in [Murphy, 2008];
- (Modal) Network Operational Semantics in [Jia and Walker, 2004] and [Park, 2006];
- (BHK-inspired) Operational Semantics of expressions encoding proofs in LP in terms of global computation in [Artemov and Bonelli, 2007];

Semantics with indexed modal types

- $a_i : \alpha$ expresses the existence of a program valid at location i of type α;
- $\Gamma_i \vdash \alpha$ is the sequence of computational steps valid at location i that validate a program of type α;
- the meaning of program α is given by explaining how steps in Γ_i are obtained and where they hold;
- Use modalities in $\circ_i \Gamma \vdash \alpha$ to express local/global validity of program/processes.

Translation to an Operational Semantics

- Provide a syntax-oriented inductively defined semantics reflecting the original BHK proof interpretation;
- Define the behavior of programs by transition relations among states of the corresponding (abstract) machine;
- Define the valid transitions as a set of inference rules to give a composite piece of syntax in terms of the transitions of its components;
- Enrich the language with locations and values/code mobility operations.

Language

Definition (Syntax of the Programming Language)

The syntax is defined by the following alphabet:

$Types := \alpha \mid \alpha \times \beta \mid \alpha \sqcup \beta \mid \alpha \rightarrow \beta \mid \alpha \supset \beta$

$Terms\ \mathcal{T} := x_i \mid a_i$

$Functions := exec(\alpha) \mid run_i(\alpha) \mid run_{i \cup j}(\alpha \cdot \beta) \mid run_{i \cap j}(\alpha \cdot \beta) \mid synchro_j(\beta(exec(\alpha)))$

$Contexts\ \mathcal{C} := \Delta_i \mid \Gamma_i \mid \circ_{i,j}\Gamma$

$Remote\ Operations := GLOB(\Box_{i \cup j}\Gamma, \alpha) \mid BROAD(\Diamond_{i \cap j}\Gamma, \alpha)$

$Portable\ Code := RET(\Gamma_{i \cup j}, \alpha) \mid SEND(\Gamma_{i \cap j}, \alpha)$

Conventions

- *exec* refers to the output of a running program; can take *any* index;
- *run* is the procedural representation of a function; occurs with a single index when referring to a single process;
- *run* takes compositions of indices when it composes processes: $\cup$ for executability at either location; $\cap$ for executability at ordered intersection;
- *synchro* computes a function using *exec* of some value it depends from (Call by Value): semantic equivalent for β-reduction or function application;
- Introduction Rules for Modalities correspond to Rules for Remote Operations; Eliminations Rules to Rules for Portable Code.

Operational Semantics

Definition (State Machine)

A state machine $S \in \mathcal{S}$

$$\mathcal{S} := (\mathcal{C}, t.i : \alpha) \mid \mathcal{C} \in \mathit{Context};\ t \in \mathcal{T}; i \in \mathcal{I}; \alpha \in \mathit{Types}$$

is an occurrence of an indexed typed term in context.

Computational Rules

Definition (Typing Rules)

$$\frac{}{\Delta_i, a_i : \alpha \vdash exec(\alpha)}\ \text{global} \qquad \frac{}{\Gamma_i, x_i : \alpha; \Delta_i \vdash run_i(\alpha)}\ \text{local}$$

$$\frac{a_i : \alpha \quad b_j : \beta}{run_{i \cup j}(\alpha \times \beta)}\ I\times \qquad \frac{a_i : \alpha}{run_i(\alpha \sqcup \beta)}\ I\sqcup$$

$$\frac{a_i : \alpha \quad exec(\alpha) \vdash b_j : \beta}{run_{i \cup j}(\alpha \to \beta)}\ I \to \qquad \frac{x_i : \alpha \quad run_i(\alpha) \vdash b_j : \beta}{run_{i \cap j}(\alpha \supset \beta)}\ I \supset$$

$$\frac{run_{i \cap j}(\alpha \supset \beta) \quad a_i : \alpha}{synchro_j(b(exec(\alpha)))}\ synchro$$

The Modal Extension

Definition (Modal Judgements)

The set of modal judgements $\mathcal{M}$ for any $i \in \mathcal{G}$ is defined by the following modal formation rules:

$$\frac{exec(\alpha)}{\Box_i \Gamma \vdash \alpha}\ \Box - \text{Formation} \qquad \frac{\Gamma_i \vdash run_i(\alpha)}{\Diamond_i \Gamma \vdash \alpha}\ \Diamond - \text{Formation}$$

Modal Rules

Definition

$$\frac{\Gamma_i, x_j : \alpha \vdash run_j(\alpha) \quad \Box_i\Gamma, x_j(a_j) : \alpha \vdash exec(\alpha)}{GLOB(\Box_{i\cup j}\Gamma, \alpha)} \; RPC1$$

$$\frac{\Gamma_i, x_j : \alpha \vdash run_j(\alpha) \quad \Diamond_i\Gamma \vdash run_j(\alpha)}{BROAD(\Diamond_{i\cap j}\Gamma, \alpha)} \; RPC2$$

$$\frac{\Box_i\Gamma, a_j : \alpha \vdash exec(\alpha) \quad GLOB(\Box_{i\cup j}\Gamma, \alpha)}{RET(\Gamma_{i\cup j}, \alpha)} \; PORT1$$

$$\frac{\Box_i\Gamma, x_j : \alpha \vdash run_{i\cap j}(\alpha) \quad BROAD(\Diamond_{i\cap j}\Gamma, \alpha)}{SEND(\Gamma_{i\cap j}, \alpha)} \; PORT2$$

Operational Semantics

Definition (Operational Model)

An indexed transition system (also called Network)

$$\textit{Networks}\ \mathcal{N} := (\mathcal{S}, \mapsto, \mathcal{I})$$

is a triple where $\mathcal{S}$ is a set of states, $\mathcal{I}$ is a set of indices and $\mapsto (\mathcal{S} \times \mathcal{I} \times \mathcal{S})$ is a ternary relation of indexed transitions. If $S, S' \in \mathcal{S}$ and $i, j \in \mathcal{I}$, then $\mapsto (S, i, j, S')$ is written as $S_i \mapsto S'_j$. This means that there is a transition $\mapsto$ from state S valid at index i to state S' valid at index j defined according to the state typing rules.

Evaluation

Rewriting rules for states transition:

	$S \mapsto S'$
run	$(\Gamma_i, x_i : \alpha) \mapsto (\Diamond_i \Gamma, run_i(\alpha))$
exec	$(\Gamma_i, a_i : \alpha) \mapsto (\Box_i \Gamma, exec(\alpha))$
$\rightarrow$	$(\Gamma_i, exec(\alpha) \vdash b_j) \mapsto (\Box_i \Gamma, run_{i \cup j}(\alpha \rightarrow \beta))$
$\supset$	$(\Gamma_i, run_i(\alpha) \vdash b_j) \mapsto \Box_i \Gamma, synchro(b_j(exec(\alpha))))$
$\times$	$(\Gamma_i, exec(\alpha), exec(\beta)) \mapsto (\Box_i \Gamma, run_{i \cup j}(\alpha \times \beta))$
$\sqcup$	$(\Gamma_i, exec(\alpha)) \mapsto (\Box_i \Gamma, run_i(\alpha \sqcup \beta))$
$\Box 1$	$(\Gamma_i, exec(\alpha)) \mapsto (GLOB(\Box_{i \cup j} \Gamma, \alpha))$
$\Box 2$	$(\Box_i \Gamma, \alpha_{i \cup j}) \mapsto (RET(\Gamma_{i \cup j}, \alpha))$
$\Diamond 1$	$(\Gamma_i, run_i(\alpha)) \mapsto (BROAD(\Diamond_{i \cap j} \Gamma, \alpha))$
$\Diamond 2$	$(\Diamond_i \Gamma, \alpha_{i \cap j}) \mapsto (SEND(\Gamma_{i \cap j}, \alpha))$

Semantic Validity

Definition (Semantic Expressions)

- Evaluation defines strong typing (normalisation) by reduction to expressions $(\Box_i\Gamma, exec(\alpha))$ and $GLOB(\Box_i\Gamma, \alpha)$.
- Expressions $(\Gamma_i, run_i(\alpha))$ and $BROAD(\Diamond_i\Gamma, \alpha)$ are admissible procedural steps but may fail to produce a safe value (when called upon at wrong addresses).
- This makes (only) the following expressions valid (safely evaluated):

$$\frac{}{a_i : \alpha \ value} \qquad \frac{}{\Box_i\Gamma, \alpha \ value}$$

Some Results

Theorem (Type Safety)

Safety is satisfied by transformations (according to the table of rewriting rules) or by terminating expression (exec(α))

1. *If* $S := (\Gamma_i, t.i : \alpha)$*, and* $S \mapsto S'$*, then* $S' := (\Gamma'_i, t'.i : \alpha)$*;*
2. *If* $S := (\Gamma_i, t.i : \alpha)$*, then either exec(*$\alpha$*) is the output value or there are* Γ', t', α' *for* $S' := (\Gamma'_i, t'.i : \alpha')$ *s.t.* $S \mapsto S'$*.*

Proof.

By (*i*) evaluation steps preserve typing. By (*ii*) closed expressions induce overall execution, hence are safe processes. □

Some Results

Theorem (Preservation)

If $S := (\Gamma_i, t.i : \alpha)$, *then* $S \mapsto S'$ *for some* $S' := (\Box\Gamma'_i, t'.i : \alpha')$.

Proof.

By induction on $\alpha, \alpha' \in$ *Types* and the structure of Γ_i and by the Safety Theorem for $S \mapsto S'$. □

Some results

Theorem (Progress)

If $S := (\Box_i\Gamma, t.i : \alpha)$*, then either* $S \mapsto S'$ *or* $exec(\alpha)$ *is the output value.*

Proof.

By induction on $\alpha \in$ *Types* using the properties induced by $\Box_i\Gamma$; by Safety Theorem for $S \mapsto S'$ and using the Preservation Theorem as last step. $\Box$

1 Conditions for Knowing

2 When Conditions are (and are not) satisfiable

3 A refiniment of constructive epistemology

One proof-object (in context), Three epistemic states

Using the crucial role of localized contextual conditions for knowledge we reconsider the schema of knowledge attitudes:

One proof-object (in context), Three epistemic states

Using the crucial role of localized contextual conditions for knowledge we reconsider the schema of knowledge attitudes:

1. proof-objects provide a precise declination of the epistemic attitude of *knowledge-that*;
2. contexts for knowledge define the epistemic attitude of *knowledge-how*;
3. extending to modalities, identifying non-terminating and locally valid processes, one articulates the attitude of *knowledge-whether*.

Knowing That

At the basis of the basic epistemic description of proof-objects is the Russellian distinction between ‘knowledge by description’ and ‘knowledge by acquaintance’ ([Russell, 1959], reformulated by Ryle as ‘knowledge-that’ and ‘knowledge-how’ [Ryle, 1949]).

Knowing That

At the basis of the basic epistemic description of proof-objects is the Russellian distinction between 'knowledge by description' and 'knowledge by acquaintance' ([Russell, 1959], reformulated by Ryle as 'knowledge-that' and 'knowledge-how' [Ryle, 1949]).

Definition

Knowledge-that amounts to knowledge of the truth of a proposition, i.e. knowledge that a proposition is true ("A is true"). The epistemic state derived by knowing-that produces justified knowledge on the basis of the related proof-object.

Knowing How

Definition

Knowledge-how corresponds to the ability of stating the truth of a certain proposition, in terms of knowledge of the set of propositions making it true. To 'know-how' A is true, means to be able to lay down a demonstration for proposition A in terms of the things one needs to know in order to know A.

Extending the dichotomy

The layers in the description of epistemic acts just illustrated can be summarized by the following structure:

1. proof-objects express the truth of a content;
2. assertion conditions of a proof-object express contextual knowledge needed by the correctness of the epistemic process;
3. contextual validity of proof-objects express local correctness of its conditions.

Extending the dichotomy

The layers in the description of epistemic acts just illustrated can be summarized by the following structure:

1. proof-objects express the truth of a content;
2. assertion conditions of a proof-object express contextual knowledge needed by the correctness of the epistemic process;
3. contextual validity of proof-objects express local correctness of its conditions.

Hence there is a third layer hidden behind the localization of conditions.

Knowing Whether

At each such description level including a proof-object *a* for content *A* corresponds a different epistemic attitude of subject *S* towards the truth of *A*:

Epistemic Act	Attitude
S knows that A is true	*S* possesses a proof-object *a* of *A*

Knowing Whether

At each such description level including a proof-object *a* for content *A* corresponds a different epistemic attitude of subject *S* towards the truth of *A*:

Epistemic Act	Attitude
S knows that A is true	*S* possesses a proof-object *a* of *A*
S knows how A is true	*S* possesses a proof-object *a* of *A* and *S* knows all the conditions needed to formulate *a*

Knowing Whether

At each such description level including a proof-object *a* for content *A* corresponds a different epistemic attitude of subject *S* towards the truth of *A*:

Epistemic Act	Attitude
S knows that A is true	*S* possesses a proof-object *a* of *A*
S knows how A is true	*S* possesses a proof-object *a* of *A* and *S* knows all the conditions needed to formulate *a*
S knows whether A is true	*S* knows how *A* is true and *S* knows where the conditions needed to formulate *a* hold

Knowing Whether

Definition

Knowledge-whether "A is true" corresponds to the ability of stating the set of propositions making A true and to lay down the contextual limits where such conditions hold.

Iterations

Is the following iteration meaningful?

S knows that S' knows that A is true

It is, but it does not allow S to know that A is true. To make it explicit we can now move to the higher description level.

Iterations

Given the distinction with "knowing-how", it is perfectly possible that, given "S' knows that A is true" it holds that

S knows that A is true, but S does not know how A is true.

Given S' knows a proof object for A, it is known to S that A is true, but not how to make A true, i.e. the subject misses the procedural aspect.

Iterations

What about the following?

> *S knows that/how A is true, but S does not know whether A is true*

Iterations

What about the following?

> *S knows that/how A is true, but S does not know whether A is true*

Seemingly not a possible iteration, but only if *knowing whether* is always taken as holding in the actual state.

Iterations

What the current interpretation of knowing-whether allows, is

S knows that/how A is true, but S does not know whether A is true (in such-and-such) conditions.

In this new sense, though *S* possesses a proof-trace that makes *A* true and has knowledge of how to execute it, *S* does not know if such proof-trace is valid under given conditions.

Iterations

Finally, the iteration of 'knowing-whether'

S knows that S' knows whether A is true

does not allow S to know whether A is true, nor it is possible for A to know that A.

Few Answers

1. YES: it is possible to formulate a theory of proof-objects fit for non-mathematical sentences;
2. YES: we want one such theory, as it allows to combine verificationism with contextualism;
3. The underlying constructive epistemology results more structured and it further allows for developing side-issues such as interaction and failure.

References I

Artemov, S. and Bonelli, E. (2007).
The intensional lambda calculus.
In *Proceedings of the international symposium on Logical Foundations of Computer Science*, LFCS '07, pages 12–25, Berlin, Heidelberg. Springer-Verlag.

Jia, L. and Walker, D. (2004).
Modal Proofs as Distributed Programs.
In *Programming Languages and Systems, ESOP2004*, volume 2986 of *Lectures Notes in Computer Science*. Springer Verlag.

Martin-Löf, P. (1993).
Philosophical aspects of intuitionistic type theory.
Unpublished notes of lectures given at the Faculteit Wijsbegeerte Leiden.

Mulligan, K., Simons, P., and Smith, B. (1984).
Truth-Makers.
Philosophy and Phenomenological Research, 44:287–321.

References II

- Murphy, T. (2008).
  Modal Types for Mobile Code.
  PhD thesis, School of Computer Science, Carnegie Mellon University.
  CMU-CS-08-126.
- Park, S. (2006).
  A modal language for the safety of mobile values.
  In *In Fourth ASIAN Symposium on Programming Languages and Systems*, pages 217–233. Springer.
- Primiero, G. (2004).
  Presuppositions, assumptions, premises.
  Master's thesis, Universiteit Leiden.
- Primiero, G. (2009a).
  An epistemic logic for becoming informed.
  Synthese (KRA), 167(2):363–389.

References III

Primiero, G. (2009b).
Epistemic modalities.
In Primiero, G. and Rahman, S., editors, *Acts of Knowledge: History, Philosophy and Logic*, volume 9 of *Tributes*, pages 207–232. College Publications.

Primiero, G. (2011).
A multi-modal type system and its procedural semantics for safe distributed programming.
In *Intuitionistic Modal Logic and Applications Workshop (IMLA11)*, Nancy.
Manuscript.

Primiero, G. (2012).
A contextual type theory with judgemental modalities for reasoning from open assumptions.
Logique & Analyse, forthcoming.

References IV

Primiero, G. (Forthcoming (2013)).
Logical validity by modal types: Information control, failure and interaction.
Logique & Analyse.

Russell, B. (1959).
The Problems of Philosophy.
Oxford University Press.

Ryle, G. (1949).
The concept of mind.
University of Chicago Press.

Sundholm, G. (2004).
Antirealism and the role of truth.
In *Handbook of Epistemology*, pages 437–466. Kluwer.

References V

van Atten, M. (to appear).
The hypothetical judgement in the history of intuitionistic logic.
In Glymour, C., W. W. W. D., editor, *Logic, Methodology, and Philosophy of Science XIII: Proceedings of the 2007 International Congress, Beijing*. London : King's College Publications.