University of Athens
Seminar Logic and Algorithms

Proofs, Evidence, Knowledge

Sergei Artemov & Elena Nogina

(*The City University of New York*)

January 20, 2006

Brouwer-Heyting-Kolmogorov (the early 1930s):

the intended provability semantics of intuitionistic logic

- a proof of $A \wedge B$ consists of a proof of A and a proof of B,
- a proof of $A \vee B$ is given by presenting either a proof of A or a proof of B,
- a proof of $A \rightarrow B$ is a construction which, given a proof of A returns a proof of B,
- absurdity $\perp$ is a proposition which has no proof, $\neg A$ is $A \rightarrow \perp$.

Gödel provability calculus (actually, the good old **S4**)

Classical axioms and rules

$\Box(F \rightarrow G) \rightarrow (\Box F \rightarrow \Box G)$ *(implicit application)*

$\Box F \rightarrow F$ *(reflexivity)*

$\Box F \rightarrow \Box\Box F$ *(implicit proof checker)*

Internalization rule:

$$\frac{\vdash F}{\vdash \Box F}$$

Reflects the basic intuition of Provability as a logic operator.

Gödel's embedding of **Int** into **S4**:

1. translate **Int**-formula F into a classical language $\Box$:

$$tr(F) = \text{“}\textit{box each subformula of } F\text{”},$$

2. test the translation in **S4**:

$$\textbf{Int}\ \textit{proves}\ F \quad\Leftrightarrow\quad \textbf{S4}\ \textit{proves}\ tr(F)$$

(Gödel (1933), McKinsey & Tarski (1948))

The mission was not accomplished though, since **S4** itself was left without an exact provability model (a natural candidate, formal provability, is not reflexive)

$$\textbf{Int} \;\hookrightarrow\; \textbf{S4} \hookrightarrow ? \;\hookrightarrow \textit{REAL PROOFS}$$

Cure (Gödel 1938): explicit proofs rather then provability

Change the format of **S4** from

$$\Box F \quad \sim \quad F \text{ is provable}$$

to

$$t{:}F \quad \sim \quad t \text{ is a proof of } F$$

This Gödel's suggestion remained unpublished until rediscovered independently in 1995.

Proof Polynomials

A basis for all invariant propositional operations on proofs

variables $x, y, z, \ldots$	*ranging over proofs*
constants $a, b, c, \ldots$	*proofs of instances of logical axioms*
"$\cdot$" is **application**:	*applies* $s{:}(F \rightarrow G)$ *to* $t{:}F$ *and returns* $(s \cdot t){:}G$
"!" is **proof checking**:	*computes* $!t$ *a proof of* $t{:}F$
"$+$" is **union**:	*takes union (concatenation) of two proofs*

Logic of Proofs

LP is the classical logic with additional atoms $p{:}F$ for proof assertions

(p is a proof polynomial and F is a formula)

A0. classical axioms

A1. $t{:}(F \rightarrow G) \ \rightarrow (s{:}F \rightarrow (t\cdot s){:}G)$ *(application)*

A2. $t{:}F \rightarrow F$ *(reflexivity)*

A3. $t{:}F \ \rightarrow\ !t{:}(t{:}F)$ *(proof checker)*

A4. $s{:}F \rightarrow (s+t){:}F, \quad t{:}F \rightarrow (s+t){:}F$ *(sum)*

R1. modus ponens

R2. $\vdash c{:}A$, *where* $A \in$*A0-A4,* c *is a proof constant.* *(constant specification)*

Another close relative: typed combinatory logic **CL**.

Combinatory terms have dual meaning as typed terms and as derivations in a Hilbert style proof system. Constant combinators stand for proofs of axioms:

$$\mathbf{k}^{A,B} : (A\rightarrow(B\rightarrow A)), \qquad \mathbf{s}^{A,B,C} : [(A\rightarrow(B\rightarrow C))\rightarrow((A\rightarrow B)\rightarrow(A\rightarrow C))]$$

Variables in **CL** denote unspecified proofs, the operation of application "$\cdot$" corresponds to the rule *modus ponens*

$$t{:}(F\rightarrow G)\ \rightarrow (s{:}F\rightarrow (t\cdot s){:}G)$$

The whole of **CL** corresponds to a fragment of **S4** consisting only of formulas of the sort

$$\Box A_1\wedge\ldots\wedge\Box A_n\rightarrow\Box B,$$

where $A_1,\ldots,A_n,B$ do not contain modalities.

Internalization in **LP** $\approx$ Curry-Howard isomorphism as a rule!

$$\frac{\vdash F}{\vdash p{:}F \quad \textit{for some proof polynomial } p}$$

$$\frac{\Gamma \vdash F}{\vec{x}{:}\Gamma \vdash t(\vec{x}){:}F \ \textit{ for some proof polynomial } t(\vec{x})}$$

Example of an **S4**-derivation converted into an **LP**-derivation.

Derivation in **S4** Derivation in **LP**

$\Box A \rightarrow \Box A \vee B$

$\Box(\Box A \rightarrow \Box A \vee B)$

$\Box\Box A \rightarrow \Box(\Box A \vee B)$

$\Box A \rightarrow \Box\Box A$

$\Box A \rightarrow \Box(\Box A \vee B)$

$B \rightarrow \Box A \vee B$

$\Box(B \rightarrow \Box A \vee B)$

$\Box B \rightarrow \Box(\Box A \vee B)$

$\Box A \vee \Box B \rightarrow \Box(\Box A \vee B)$

Example of an **S4** derivation converted into an **LP**-derivation.

Derivation in **S4**	Derivation in **LP**
$\Box A \to \Box A \vee B$	$x{:}A \to x{:}A \vee B$
$\Box(\Box A \to \Box A \vee B)$	$a{:}(x{:}A \to x{:}A \vee B)$
$\Box\Box A \to \Box(\Box A \vee B)$	$!x{:}x{:}A \to (a\cdot !x){:}(x{:}A \vee B)$
$\Box A \to \Box\Box A$	$x{:}A \to !x{:}x{:}A$
$\Box A \to \Box(\Box A \vee B)$	$x{:}A \to (a\cdot !x){:}(x{:}A \vee B)$
$B \to \Box A \vee B$	
$\Box(B \to \Box A \vee B)$	
$\Box B \to \Box(\Box A \vee B)$	
$\Box A \vee \Box B \to \Box(\Box A \vee B)$	

Example of an **S4** derivation converted into an **LP**-derivation.

Derivation in **S4**	Derivation in **LP**
$\Box A \to \Box A \vee B$	$x{:}A \to x{:}A \vee B$
$\Box(\Box A \to \Box A \vee B)$	$a{:}(x{:}A \to x{:}A \vee B)$
$\Box\Box A \to \Box(\Box A \vee B)$	$!x{:}x{:}A \to (a\cdot !x){:}(x{:}A \vee B)$
$\Box A \to \Box\Box A$	$x{:}A \to !x{:}x{:}A$
$\Box A \to \Box(\Box A \vee B)$	$x{:}A \to (a\cdot !x){:}(x{:}A \vee B)$
$B \to \Box A \vee B$	$B \to x{:}A \vee B$
$\Box(B \to \Box A \vee B)$	$b{:}(B \to x{:}A \vee B)$
$\Box B \to \Box(\Box A \vee B)$	$y{:}B \to (b\cdot y){:}(x{:}A \vee B)$
$\Box A \vee \Box B \to \Box(\Box A \vee B)$	

Example of an **S4** derivation converted into an **LP**-derivation.

Derivation in **S4**	Derivation in **LP**
$\Box A \to \Box A \lor B$	$x{:}A \to x{:}A \lor B$
$\Box(\Box A \to \Box A \lor B)$	$a{:}(x{:}A \to x{:}A \lor B)$
$\Box\Box A \to \Box(\Box A \lor B)$	$!x{:}x{:}A \to (a\cdot !x){:}(x{:}A \lor B)$
$\Box A \to \Box\Box A$	$x{:}A \to !x{:}x{:}A$
$\Box A \to \Box(\Box A \lor B)$	$x{:}A \to (a\cdot !x){:}(x{:}A \lor B)$
$B \to \Box A \lor B$	$B \to x{:}A \lor B$
$\Box(B \to \Box A \lor B)$	$b{:}(B \to x{:}A \lor B)$
$\Box B \to \Box(\Box A \lor B)$	$y{:}B \to (b\cdot y){:}(x{:}A \lor B)$
$\Box A \lor \Box B \to \Box(\Box A \lor B)$	???

Example of an **S4** derivation converted into an **LP**-derivation.

Derivation in **S4**	Derivation in **LP**
$\Box A \to \Box A \vee B$	$x{:}A \to x{:}A \vee B$
$\Box(\Box A \to \Box A \vee B)$	$a{:}(x{:}A \to x{:}A \vee B)$
$\Box\Box A \to \Box(\Box A \vee B)$	$!x{:}x{:}A \to (a\cdot !x){:}(x{:}A \vee B)$
$\Box A \to \Box\Box A$	$x{:}A \to !x{:}x{:}A$
$\Box A \to \Box(\Box A \vee B)$	$x{:}A \to (a\cdot !x){:}(x{:}A \vee B)\ [\to (a\cdot !x + b\cdot y){:}(x{:}A \vee B)]$
$B \to \Box A \vee B$	$B \to x{:}A \vee B$
$\Box(B \to \Box A \vee B)$	$b{:}(B \to x{:}A \vee B)$
$\Box B \to \Box(\Box A \vee B)$	$y{:}B \to (b\cdot y){:}(x{:}A \vee B)\ [\to (a\cdot !x + b\cdot y){:}(x{:}A \vee B)]$
$\Box A \vee \Box B \to \Box(\Box A \vee B)$	$x{:}A \vee y{:}B \to (a\cdot !x + b\cdot y){:}(x{:}A \vee B)$

Realization Theorem (S.A.):

S4 *proves* F *iff there is an assignment* r *of proof polynomials to all* $\Box$*'s in* F *such that the corresponding realization* F^r *is derivable in* **LP***.*

Complexity of realization: A polynomial realization algorithm was suggested by Brezhnev & Kuznetz. It produces proof polynomials of at most quadratic size in the length of the given cut-free derivation in **S4**.

Other modal logics

Similar explicit versions were found for modal logics

S5 (S.A., Kazakov, & Shapiro);

K, **K4**, **T**, **D**, and **D4** (Brezhnev).

Comparing formats

Type (logic) derivation $A \to B,\ A \vdash B$

(plain types - propositions)

λ-derivation (Curry-Howard) $s{:}(A \to B),\ t{:}A \vdash (s \cdot t){:}B$

(plain typed λ-terms, explicit, but no proof iterations allowed)

Modal derivation (in **S4**) $\Box A \vee \Box B \vdash \Box(\Box A \vee \Box B)$

(provability iterates, but is implicit)

Proof polynomial derivation $x{:}A \vee y{:}B \vdash (a \cdot !x + b \cdot !y){:}(x{:}A \vee y{:}B)$

(provability is explicit $a{:}(x{:}A \to x{:}A \vee y{:}B)$

and iterates freely) $b{:}(y{:}B \to x{:}A \vee y{:}B)$

Completeness Theorem for **LP** (S.A.):

LP *derives all identities in its own language valid with respect to the semantics of formal proofs.*

Corollary: *answers both Gödel's and BHK questions*

The foundational picture now looks:

$$\textbf{Int} \quad \hookrightarrow \quad \textbf{S4} \quad \hookrightarrow \quad \textbf{LP} \quad \hookrightarrow \quad \textit{REAL PROOFS}$$

and all these embedding are exact.

Formal provability model: a good old logic **GL**.

Classical axioms and rules

$\Box(F\to G)\to(\Box F\to\Box G)$ *(implicit application)*

$\Box F\to\Box\Box F$ *(implicit proof checker)*

$\Box(\Box F\to F)\to\Box F$ *(Löb axiom)*

Necessitation rule:

$$\frac{\vdash F}{\vdash \Box F}$$

Complete with respect to interpretation $\Box F$ *as* Provable(F) *(Solovay, 1976).*

Represents incompleteness theorem, applications in Proof Theory.

Logic of Proofs and Provability: what to expect?

Negative introspection $\neg(\Box A)\rightarrow\Box\neg(\Box A)$ does not hold for Provability.

Explicit negative introspection $\neg(x:A) \rightarrow p:\neg(x:A)$ does not hold for any specific p. Indeed, suppose this holds in arithmetic for all instances of p, x, A. Fix interpretations of p, x, then the principle should hold for all A's which indicates that p is a proof of infinitely many different instances of $\neg(x{:}A)$. A contradiction.

It takes a blend of explicit and implicit to get a valid principle: introspection of explicit negative $\neg(x{:}A)\rightarrow\Box\neg(x{:}A)$.

Logic of Proofs and Provability: the system

Prototype systems have been studied since 1994 by S.A., Nogina

Sidon-Yavorskaya found a system, **LPP** , containing both **LP** and **GL**.

The minimal system **GLA** containing **LP** and **GL** was suggested by Nogina:

GLA= **GL** + **LP** + Principles connecting explicit and formal provability:

C1. $t{:}F \rightarrow \Box F$ (*explicit-implicit connection*)

C2. $\neg(t{:}F) \rightarrow \Box\neg(t{:}F)$ (*negative introspection*)

C3. $t{:}\Box F \rightarrow F$ (*weak reflexivity*)

Main theorems about **GLA** (Nogina): arithmetical completeness, internalization property, model completeness.

Examples of new principles

1. For all t, F there is a proof polynomial $\Uparrow(x)$ such that
$$\textbf{GLA} \vdash t{:}F \rightarrow \Uparrow(t){:}\Box F\ .$$

$a{:}(t{:}F \rightarrow \Box F)$

$!t{:}t{:}F \rightarrow (a\cdot !t){:}\Box F$

$t{:}F \rightarrow (a\cdot !t){:}\Box F$, and we can pick $\Uparrow(x) = a\cdot !x$

2. For all t, F there is a proof polynomial $\Downarrow(x)$ such that
$$\textbf{GLA} \vdash t{:}\Box F \rightarrow \Downarrow(t){:}F\ .$$

$b{:}(t{:}\Box F \rightarrow F)$

$!t{:}t{:}\Box F \rightarrow (b\cdot !t){:}F$

$t{:}\Box F \rightarrow (b\cdot !t){:}F$, and we can pick $\Downarrow(x) = b\cdot !x$

Explicit Löb's Theorem

For all F there is a proof polynomial $l(x)$ such that

$$\textbf{GLA} \vdash x{:}(\Box F \to F) \to \ l(x){:}F \ .$$

$c{:}(\Box(\Box F \to F) \to \Box F)$

$y{:}\Box(\Box F \to F) \to (c \cdot y){:}\Box F$

$x{:}(\Box F \to F) \to \Uparrow(x){:}\Box(\Box F \to F)$, and we can set $y = \Uparrow(x)$

$x{:}(\Box F \to F) \to \Downarrow(c \cdot \Uparrow(x)){:}F$, and we can set $l(x) = \Downarrow(c \cdot \Uparrow(x))$

Existential semantics for modal logic.

Gödel, 1933,38: $\Box F$ was read as

there exists a proof (witness) for F.

Existential understanding of modality is also typical of "naive" semantics for a wide range of epistemic and provability logics. Formal provability semantics is an example, which works for **GL** but does not work for **S4**. Existential semantics for the major modal logic **S4** was not formalized until proof polynomials.

Kripke, late 1950s: $\Box F$ is read as

in all possible situations F *holds.*

Universal semantics naturally appears in dynamic and temporal logics, computational processes.

Other developments: foundations of the logic of proofs

0. Basic Logic of Proofs = operation free versions for major classes of proof predicates, pre-**LP** studies. S.A. & T. Strassen;

1. First order logic of proofs. S.A., Yavorskaya, Yavorsky;

2. Joint Logic of Proofs and Provability. S.A., Yavorskaya, Nogina;

3. Functional Logic of Proofs. S.A. & Strassen; Krupski, Yavorskaya, Rubtsova;

4. Intuitionistic logic of proofs. S.A. and Iemhoff;

5. Logic of proofs for bounded arithmetic. Goris;

6. Logic with quantifiers over proofs. Yavorsky, Fitting;

7. Models for **LP**. Mkrtychev, Fitting, S.A.;

8. Complexity of Logic of Proofs. Kuznets, Brezhnev, Milnikel;

9. Interpolation property of **LP**. Yavorskaya;

10. Disjunctive Property, complexity of the reflexive fragment. N. Krupski;

11. Tableau and matrix proof systems for **LP**. Renne, Fitting, Bryukhov.

Other developments: applications of the logic of proofs

12. Essential self-referentiality of modal logic via logic of proofs. Kuznets;
13. Logical models of referential data structures. Krupski, S.A.;
14. Explicit provability in verification theory. S.A.;
15. Law and evidence via **LP**. Beklemishev;
16. Reflexive Combinatory Logic. S.A. & N. Krupski;
17. Reflexive lambda-calculus. S.A., Alt, Bonelli;
18. Explicit counterpart of **S5**. S.A.,Kazakov & Shapiro; Rubtsova, Pacuit;
19. Explicit counterparts of major modal logics **T**, **D**, **D4**, **K**, **K4**. Brezhnev;
20. Epistemic Logic with justification. S.A. and Nogina;
21. Evidence-based common knowledge. S.A., Antonakos;
22. **LP**, justified knowledge provers. Bryukhov.

In the rest of the talk we will speak on the epistemic logic with justification and Reflexive Combinatory Logic.

Introducing Justification into Formal Epistemology

Plato's celebrated tripartite definition of knowledge as *justified true belief* (JTB) is generally regarded as a set of necessary conditions for the possession of knowledge. Due to Hintikka, the "true belief" components have been fairly formalized by means of modal logic and its possible worlds semantics. Despite the fact that the justification condition has received the greatest attention in epistemology, it lacked a formal representation. We introduce justification into formal epistemology by combining Hintikka-style epistemic modal logic with justification calculi arising from the logic of proofs.

LP in not logically omniscient

From the epistemic point of view, the usual modal logic (say **S4**) is *logically omniscient* in the following sense: **S4** can prove epistemic assertions $\Box F$ of manageable length which are not actually true since there are no (unless NP=coNP) feasible proofs of the validity of F. Example: F is a short tautology without a feasible propositional proof.

LP passes this Non Logical Omniscience (NLO) test. A recent result by Kuznets, 2005 has shown that for any propositional formula F, **LP** is able to prove an epistemic assertion $t{:}F$ only when there is a proof of F not longer than some linear function of $\mathtt{length}(t{:}F)$.

Epistemic Logic with Justification

S4LP= **S4** + **LP** + $t{:}F \rightarrow \Box F$

Main theorems about **S4LP**. S.A. & Nogina, Fitting, 2004-05: internalization, soundness and completeness with respect to epistemic models.

S4LPN= **S4LP** + *negative introspection* $\neg(x{:}A) \rightarrow \Box\neg(x{:}A)$.

Main theorems about **S4LPN**. S.A. & Nogina, 2004-05: internalization, soundness and completeness with respect to epistemic models.

Evidence-Based Knowledge (*EBK*) systems: a case study.

Consider n copies of modal logics representing knowledge operators of n agents, $\mathsf{K}_1, \ldots, \mathsf{K}_n$, along with a systems of evidence assertions taken from the logic of proofs **LP**, i.e. proof polynomials. Here we consider representative examples.

$\mathbf{T}_n\mathbf{LP} = \mathbf{T}_n + \mathbf{LP} + t{:}\varphi \rightarrow \mathsf{K}_i\varphi$.

$\mathbf{S4}_n\mathbf{LP} = \mathbf{S4}_n + \mathbf{LP} + t{:}\varphi \rightarrow \mathsf{K}_i\varphi$.

$\mathbf{S5}_n\mathbf{LP} = \mathbf{S5}_n + \mathbf{LP} + t{:}\varphi \rightarrow \mathsf{K}_i\varphi$.

Trial schema, mathematics: a growing system of admissible evidence accepted by all the participants.

Internalization holds for *EBK*-systems.

Evidence-Based Knowledge is a special case of Common Knowledge: *Each of the evidence operators* $t{:}\varphi$ *satisfies the Fixed-Point Axiom for common knowledge.*

Evidence-Based Knowledge does not suffer the logical omniscience sickness. *One cannot claim an evidence-based knowledge of* φ *without actually building a evidence term of* φ*. Proofs are the principal protection against logical omniscience!*

Epistemic semantics for logics with justification

1994-1997. Arithmetical provability semantics and Kripke-style models for joint logics of proofs and provability (S.A., Nogina, Yavorskaya.)

1997-2003. Mkrtychev symbolic models for **LP**, where the evidence function have been first introduced

2003. Fitting (then new in the **LP**-area) rediscovered an evidence function and introduced Kripke structure into models of **LP**. The key definition of truth value of $t{:}\varphi$ now looks $u \models t{:}\varphi$ iff

$v \models \varphi$ for all v's accesible from u, and t is an admissable evidence for φ

Later Fitting models were adapted for epistemic logic with justification. The resulting models cover all known epistemic systems with justification: **LP**, **GLA**, **LPP**, **S4LP**, **S4LPN**, **T**$_n$**LP**, **S4**$_n$**LP**, **S5**$_n$**LP**.

Models. A *frame* is $(W, R_1, \ldots, R_n, R)$, where $R_1, \ldots, R_n$ are reflexive (for **T**$_n$**LP**), reflexive and transitive for **S4**$_n$**LP**, etc.; R is an arbitrary reflexive and transitive relation that contains all R_i's. The idea is that $u \Vdash t{:}\varphi$ iff φ holds and has admissible evidences in all worlds R-accessible from u.

Possible evidence function $\mathcal{E}$ is a mapping from states and evidence terms to sets of formulas such that the following very natural Mkrtychev-Fitting conditions are met:

- *Monotonicity*: uRv implies $\mathcal{E}(u,t) \subseteq \mathcal{E}(v,t)$.

- *Natural closure conditions*
  Application: $\varphi \!\rightarrow\! \psi \in \mathcal{E}(u,s)$ and $\varphi \in \mathcal{E}(u,t)$ implies $\psi \in \mathcal{E}(u,s{\cdot}t)$
  Inspection: $\phi \in \mathcal{E}(u,t)$ implies $t{:}\varphi \in \mathcal{E}(u,!t)$.
  Sum: $\mathcal{E}(u,s) \cup \mathcal{E}(u,t) \subseteq \mathcal{E}(u,s+t)$.

$\Vdash$ is an arbitrary mapping from propositional letters to subsets of W.

Given a model $\mathcal{M} = (W, R_1, \ldots, R_n, R, \mathcal{E}, \Vdash)$, a forcing relation $\Vdash$ is extended from propositional letters to all formulas classically and by the usual modal rule: "$u \Vdash \mathsf{K}_i\varphi$ iff $v \Vdash \varphi$, for every $v \in W$ with uR_iv," plus

$$u \Vdash t{:}\varphi \text{ iff } \varphi \in \mathcal{E}(u,t) \text{ and } v \Vdash \varphi, \text{ for every } v \in W \text{ with } uRv.$$

A *constant specification* is a mapping $\mathcal{CS}$ from evidence constants to sets of axioms, possibly empty. A model $\mathcal{M}$ meets a constant specification $\mathcal{CS}$ if $a{:}\varphi$ holds in $\mathcal{M}$ for all a and φ such that $\varphi \in \mathcal{CS}(a)$.

S.A., 2005: **T$_n$LP**, **S4$_n$LP**, **S5$_n$LP** are sound and complete with respect to corresponding classes of Fitting models.

Common Knowledge in a nutshell.

Let $K_1, K_2, \ldots, K_n$ stand for knowledge operators in an n-agent logic of knowledge and $E\varphi \; = \; K_1\varphi \wedge K_2\varphi \wedge \ldots \wedge K_n\varphi$. The informal description of the common knowledge operator C corresponding to $K_1, K_2, \ldots, K_n$ is

$$C\varphi \;\; \Leftrightarrow \varphi \wedge E\varphi \wedge E^2\varphi \wedge E^3\varphi \ldots$$

In this "specification" the part $\Rightarrow$ represents an infinite series of axioms

$$C\varphi \;\; \rightarrow \varphi, \; C\varphi \;\; \rightarrow E\varphi, \; C\varphi \;\; \rightarrow E^3\varphi, \; \ldots.$$

To capture the part $\Leftarrow$ some extra work should be done.

In a Kripke-style model for $\mathsf{K}_1, \mathsf{K}_2, \ldots, \mathsf{K}_n$ the common knowledge operator is defined as the modality of reachability along paths that use accessibility edges corresponding to any of $\mathsf{K}_1, \mathsf{K}_2, \ldots, \mathsf{K}_n$.

Axiomatically C is defined by a very intuitive *Fixed-Point Axiom (FPA)*

$$\mathsf{C}\varphi \;\leftrightarrow\; \mathsf{E}(\varphi \wedge \mathsf{C}\varphi)$$

along with a non-intuitive *Induction Rule (IR)*

$$\frac{\varphi \to \mathsf{E}(\psi \wedge \varphi)}{\varphi \to \mathsf{C}\psi}.$$

FPA and *IR* together express the fact that C is a normal modality of the same kind as K_i, and that C corresponds to the largest solution of *FPA*.

The latter is not actually needed in paradigmatic examples normally used to illustrate the notion of Common Knowledge.

Forgetful evidence-based knowledge - a new approach to Common Knowledge: a stronger modality, but leaner axiomatic systems.

New modal operator $\mathsf{J}\varphi$ (φ *is justified*) is the forgetful projection of $t{:}\varphi$.

$\mathbf{T}_n^{\mathsf{J}}$ $=\mathbf{T}_n$ + **S4** (with modality J) + $\mathsf{J}\varphi\rightarrow\mathsf{K}_i\varphi$, $i=1,2,\ldots,n$.

$\mathbf{S4}_n^{\mathsf{J}}$ $=\mathbf{S4}_n$ + **S4** (with modality J) + $\mathsf{J}\varphi\rightarrow\mathsf{K}_i\varphi$, $i=1,2,\ldots,n$.

$\mathbf{S5}_n^{\mathsf{J}}$ $=\mathbf{S5}_n$ + **S4** (with modality J) + $\mathsf{J}\varphi\rightarrow\mathsf{K}_i\varphi$, $i=1,2,\ldots,n$.

In $\mathbf{S4}_n^{\mathsf{J}}$ the dummy $(n+1)$st modality J is exactly McCarthy's "any fool knows" modality (introduced in 1979), which now receives an exact *EBK*-semantics. J plays the role of a sceptical agent who accepts facts only if they are supplied with checkable evidence. On the other hand, this agent is trusted by all other agents and is capable of internalizing and inspecting of any fact actually proven in the system.

Realization Theorem: *$\mathbf{T}_n^{\mathsf{J}}$, $\mathbf{S4}_n^{\mathsf{J}}$ and $\mathbf{S5}_n^{\mathsf{J}}$ are exactly the forgetful projections of $\mathbf{T}_n\mathbf{LP}$, $\mathbf{S4}_n\mathbf{LP}$, and $\mathbf{S5}_n\mathbf{LP}$*, respectively.

Proof: Cut-elimination in $\mathbf{T}_n^{\mathsf{J}}$ and $\mathbf{S4}_n^{\mathsf{J}}$ followed by the realization algorithm. Fitting semantical method for $\mathbf{S5}_n^{\mathsf{J}}$.

In particular, given any principle φ provable in any of forgetful *EBK*-systems $\mathbf{T}_n^{\mathsf{J}}$, $\mathbf{S4}_n^{\mathsf{J}}$ or $\mathbf{S5}_n^{\mathsf{J}}$ one could find a realization of all occurrences of the forgetful modality J in φ such that the resulting *EBK*-formula φ^r is derivable in $\mathbf{T}_n\mathbf{LP}$, $\mathbf{S4}_n\mathbf{LP}$ or $\mathbf{S5}_n\mathbf{LP}$ respectively.

In a view of the realization algorithms, $\mathsf{J}\varphi$ can be understood as

there is an access to an evidence for φ.

Forgetful evidence-based knowledge J satisfies *FPA*

$$\mathsf{J}\varphi \;\rightarrow\; \mathsf{E}(\varphi \wedge \mathsf{J}\varphi).$$

Comparisons

Common Knowledge C	*Forgetful EBK* J
$\mathsf{C}\varphi \ \leftrightarrow \varphi \wedge \mathsf{E}\varphi \wedge \mathsf{E}^2\varphi \wedge \mathsf{E}^3\varphi \ldots$	$\mathsf{J}\varphi \ \rightarrow \varphi \wedge \mathsf{E}\varphi \wedge \mathsf{E}^2\varphi \wedge \mathsf{E}^3\varphi \ldots$
reachability along $\mathsf{K}_1, \mathsf{K}_2, \ldots, \mathsf{K}_n$	a transitive and reflexive extension of reachability
largest solution of *FPA* + normal modality principles	any solution of *FPA* + normal modality principles

$$t{:}\varphi \ \rightarrow \ \mathsf{J}\varphi \ \rightarrow \ C\varphi$$

$(\mathbf{S4}_n^{\mathsf{J}})^* \ \subset \ \mathbf{S4}_n^{\mathsf{C}}$, where $*$ is renaming C to J .

Antonakos, 2005: $\mathbf{S4}_n^{\mathsf{C}} = (\mathbf{S4}_n^{\mathsf{J}})^* + (\varphi \wedge \mathsf{C}(\varphi \rightarrow \mathsf{E}\varphi) \rightarrow \mathsf{C}\varphi)$.

Advantages of *EBK*-systems vs. the usual Common Knowledge

1. More practical and automation friendly. Forgetful *EBK*-systems may be easier to work with, since they are the standard modal logics supported by a well-developed machinery. Cut elimination opens a way to automated proof search methods.

2. More tractable. The question of whether a given real system has an evidence-based knowledge can be reduced to checking a manageable set of model-independent conditions.

3. More flexible. Evidence-based knowledge provides an additional degree of flexibility, since an evidence part can be chosen independently of knowledge systems of individual agents. The common knowledge operator is a derivative of the agent knowledge operators and carries the features of the latter.

4. Omniscience-free. An agent cannot claim to have evidence-based knowledge without have actually built a supporting evidence term.

Extensional vs. intensional knowledge representation

There is one more issue which is naturally handled in the epistemic logic with justification: an intensional and extensional representation of knowledge. Knowledge statements "F is known" ($\Box F$) remain *extensional*, as in Hintikka's logic of knowledge, whereas new justification statements $t{:}F$ are already *intensional*. Indeed, the facts that $t{:}F$ holds and G is (even provably) equivalent to F do not yield $t{:}G$ as well. If there is a justification s for $F \rightarrow G$, then a justification for G is a certain function of s and t, which is, generally speaking, different from t. Formal axioms and rules of epistemic logic with justification capture this distinction.

Reflexive Combinators

Characteristic features of the Reflexive Combinatory Logic **RCL** are the Combinatory Logic format, a Church style rigid typing, the implicational intuitionistic logic on level 0, and the Internalization Property, which immediately captures the usual **CL** and much more.

$\mathbf{k}{:}[A\to(B\to A)]$	*old combinator* **k**
$\mathbf{s}{:}[(A\to(B\to C))\to((A\to B)\to(A\to C))]$	*old combinator* **s**
$\mathbf{d}{:}[t{:}F \to F]$	*DENOTATE*
$\mathbf{o}{:}[t{:}(F \to G) \to (s{:}F \to (t\cdot s){:}G)]$	*COMPUTOR*
$\mathbf{c}{:}[t{:}F \to\ !t{:}(t{:}F)]$	*CODING*

Computational semantics.

Standard set theoretical semantics of types, e.g. functional types are interpreted as sets of total functions. Some of the objects have constructive counterparts *names*, e.g. functions - programs that compute them. $t{:}F$ is interpreted as a name (program) of type F. A more pedantic eye should already figure out that $t{:}F$ is rather a singleton, i.e. a single element set containing the name (program) above.

d:$[t{:}F \to F]$ - realizes a fundamental denotational correspondence *name - object*, in particular, *program - function*.

o:$[t{:}(F \to G) \;\to (s{:}F \to (t{\cdot}s){:}G)]$ represents a computor, which maps a program t and an input s to the result $t \cdot s$

c:$[t{:}F \;\to\, !t{:}(t{:}F)]$ maps a program into its code (alias, name, etc.). Examples: t is a bytecode of a function, $!t$ - its ML code, $!!t$ its higher level code with an interpreter to ML, $!!!t$ - its file name (something like *deepblue7-12.exe*), etc.