On Universal Cognitive Intelligence (UCI), Briefly

Selmer Bringsjord

Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science
Department of Computer Science
Lally School of Management & Technology
Rensselaer Polytechnic Institute (RPI)
Troy, New York 12180 USA

Intro to Logic-based AI
10/28/2024

Core Ideas @ Inception, & Context for Great Computational Intelligence … Which Led to Universal Cognitive Intelligence

In general, for a computational artifact $\mathcal{C}$ to have GCI, we hold that it must produce a result ρ that is,

Significant by at least near-consensus among relevant humans, intrinsically significant;

Independent generated by a problem-solving run carried out to a high degree by $\mathcal{C}$ independent of human insight and assistance; and

Innovative where this problem-solving run begins from a starting point ι that is a "long distance" from ρ.

We shall assume that λ applied to a pair (ι, ρ) yields a distance δ; we therefore write

$$\lambda(\iota, \rho) = \delta.$$

To say that $\mathcal{C}$ produces ρ having started with ι, we write

$$\mathcal{C} : \iota \longrightarrow \rho.$$

We shall further assume that the general space of inputs is $\iota^\star$, and the general space of results $\rho^\star$. Under this notation, it can be informatively said that a good indicator of whether a result is significant is that the function f from $\iota^\star$ to $\rho^\star$ is Turing-unsolvable. Were this indicator promoted to an absolute requirement, which is quite tempting, the first property of GCI could plausibly be formalized via something like the following equation as a necessary condition for this property (significance) to be possessed.[7]

$$\mathcal{C} : \iota \longrightarrow \rho \text{ where the function } f : \iota^\star \longrightarrow \rho^\star \text{ is Turing-unsolvable.} \quad (2)$$

[7] One must be careful here. Let h be a binary halting function taking as input the Gödel number n^M of a Turing machine M along with input m to that Turing machine. As is well-known, h is Turing-uncomputable. Yet there are individual Turing machines, accompanied by inputs to them, which can be instantly declared and proved to be either

In general, for a computational artifact $\mathcal{C}$ to have GCI, we hold that it must produce a result ρ that is,

Significant by at least near-consensus among relevant humans, intrinsically significant;

Independent generated by a problem-solving run carried out to a high degree by $\mathcal{C}$ independent of human insight and assistance; and

Innovative where this problem-solving run begins from a starting point ι that is a "long distance" from ρ.

We shall assume that λ applied to a pair (ι, ρ) yields a distance δ; we therefore write

$$\lambda(\iota, \rho) = \delta.$$

To say that $\mathcal{C}$ produces ρ having started with ι, we write

$$\mathcal{C} : \iota \longrightarrow \rho.$$

We shall further assume that the general space of inputs is $\iota^\star$, and the general space of results $\rho^\star$. Under this notation, it can be informatively said that a good indicator of whether a result is significant is that the function f from $\iota^\star$ to $\rho^\star$ is Turing-unsolvable. Were this indicator promoted to an absolute requirement, which is quite tempting, the first property of GCI could plausibly be formalized via something like the following equation as a necessary condition for this property (significance) to be possessed.[7]

$$\mathcal{C} : \iota \longrightarrow \rho \text{ where the function } f : \iota^\star \longrightarrow \rho^\star \text{ is Turing-unsolvable.} \quad (2)$$

[7] One must be careful here. Let h be a binary halting function taking as input the Gödel number n^M of a Turing machine M along with input m to that Turing machine. As is well-known, h is Turing-uncomputable. Yet there are individual Turing machines, accompanied by inputs to them, which can be instantly declared and proved to be either

In general, for a computational artifact $\mathcal{C}$ to have GCI, we hold that it must produce a result ρ that is,

Significant by at least near-consensus among relevant humans, intrinsically significant;

Independent generated by a problem-solving run carried out to a high degree by $\mathcal{C}$ independent of human insight and assistance; and

Innovative where this problem-solving run begins from a starting point ι that is a "long distance" from ρ.

We shall assume that λ applied to a pair (ι, ρ) yields a distance δ; we therefore write

$$\lambda(\iota, \rho) = \delta.$$

To say that $\mathcal{C}$ produces ρ having started with ι, we write

We'd said: "Even famous AI systems strike out." — pre AlphaGo.

$$\mathcal{C} : \iota \longrightarrow \rho.$$

We shall further assume that the general space of inputs is $\iota^\star$, and the general space of results $\rho^\star$. Under this notation, it can be informatively said that a good indicator of whether a result is significant is that the function f from $\iota^\star$ to $\rho^\star$ is Turing-unsolvable. Were this indicator promoted to an absolute requirement, which is quite tempting, the first property of GCI could plausibly be formalized via something like the following equation as a necessary condition for this property (significance) to be possessed.[7]

$$\mathcal{C} : \iota \longrightarrow \rho \text{ where the function } f : \iota^\star \longrightarrow \rho^\star \text{ is Turing-unsolvable.} \quad (2)$$

[7] One must be careful here. Let h be a binary halting function taking as input the Gödel number n^M of a Turing machine M along with input m to that Turing machine. As is well-known, h is Turing-uncomputable. Yet there are individual Turing machines, accompanied by inputs to them, which can be instantly declared and proved to be either

In general, for a computational artifact $\mathcal{C}$ to have GCI, we hold that it must produce a result ρ that is,

Significant by at least near-consensus among relevant humans, intrinsically significant;

Independent generated by a problem-solving run carried out to a high degree by $\mathcal{C}$ independent of human insight and assistance; and

Innovative where this problem-solving run begins from a starting point ι that is a "long distance" from ρ.

We shall assume that λ applied to a pair (ι, ρ) yields a distance δ; we therefore write

$$\lambda(\iota, \rho) = \delta.$$

To say that $\mathcal{C}$ produces ρ having started with ι, we write

$$\mathcal{C} : \iota \longrightarrow \rho.$$

We'd said: "The result must be provably correct (even when won on the strength of *in*ductive reasoning)."

We shall further assume that the general space of inputs is $\iota^\star$, and the general space of results $\rho^\star$. Under this notation, it can be informatively said that a good indicator of whether a result is significant is that the function f from $\iota^\star$ to $\rho^\star$ is Turing-unsolvable. Were this indicator promoted to an absolute requirement, which is quite tempting, the first property of GCI could plausibly be formalized via something like the following equation as a necessary condition for this property (significance) to be possessed.[7]

$$\mathcal{C} : \iota \longrightarrow \rho \text{ where the function } f : \iota^\star \longrightarrow \rho^\star \text{ is Turing-unsolvable.} \quad (2)$$

[7] One must be careful here. Let h be a binary halting function taking as input the Gödel number n^M of a Turing machine M along with input m to that Turing machine. As is well-known, h is Turing-uncomputable. Yet there are individual Turing machines, accompanied by inputs to them, which can be instantly declared and proved to be either

In general, for a computational artifact $\mathcal{C}$ to have GCI, we hold that it must produce a result ρ that is,

Significant by at least near-consensus among relevant humans, intrinsically significant;

Independent generated by a problem-solving run carried out to a high degree by $\mathcal{C}$ independent of human insight and assistance; and

Innovative where this problem-solving run begins from a starting point ι that is a "long distance" from ρ.

We shall assume that λ applied to a pair (ι, ρ) yields a distance δ; we therefore write

$$\lambda(\iota, \rho) = \delta.$$

To say that $\mathcal{C}$ produces ρ having started with ι, we write

$$\mathcal{C} : \iota \longrightarrow \rho.$$

We shall further assume that the general space of inputs is $\iota^\star$, and the general space of results $\rho^\star$. Under this notation, it can be informatively said that a good indicator of whether a result is significant is that the function f from $\iota^\star$ to $\rho^\star$ is Turing-unsolvable. Were this indicator promoted to an absolute requirement, which is quite tempting, the first property of GCI could plausibly be formalized via something like the following equation as a necessary condition for this property (significance) to be possessed.[7]

$$\mathcal{C} : \iota \longrightarrow \rho \text{ where the function } f : \iota^\star \longrightarrow \rho^\star \text{ is Turing-unsolvable.} \tag{2}$$

[7] One must be careful here. Let h be a binary halting function taking as input the Gödel number n^M of a Turing machine M along with input m to that Turing machine. As is well-known, h is Turing-uncomputable. Yet there are individual Turing machines, accompanied by inputs to them, which can be instantly declared and proved to be either

"Classic" ADR Result

Analogico-Deductive Generation of Gödel's First Incompleteness Theorem from the Liar Paradox

John Licato, Naveen Sundar Govindarajulu, Selmer Bringsjord, Michael Pomeranz, Logan Gittelson
Rensselaer Polytechnic Institute
Troy, NY
{licatj,govinn,selmer,pomerm,gittel}@rpi.edu

Abstract

Gödel's proof of his famous first incompleteness theorem (**G1**) has quite understandably long been a tantalizing target for those wanting to engineer impressively intelligent computational systems. After all, in establishing **G1**, Gödel did something that by any metric must be classified as stunningly intelligent. We observe that it has long been understood that there is some sort of analogical relationship between the Liar Paradox (**LP**) and **G1**, and that Gödel himself appreciated and exploited the relationship. Yet the exact nature of the relationship has hitherto not been uncovered, by which we mean that the following question has not been answered: Given a description of **LP**, and the suspicion that it may somehow be used by a suitably programmed computing machine to find a proof of the incompleteness of Peano Arithmetic, can such a machine, provided this description as input, produce as output a complete and verifiably correct proof of **G1**? In this paper, we summarize engineering that entails an affirmative answer to this question. Our approach uses what we call *analogico-deductive reasoning* (ADR), which combines analogical and deductive reasoning to produce a full deductive proof of **G1** from **LP**. Our engineering uses a form of ADR based on our META-R system, and a connection between the Liar Sentence in **LP** and Gödel's Fixed Point Lemma, from which **G1** follows quickly.

1 Introduction

Gödel's proofs of his incompleteness theorems are among the greatest intellectual achievements of the 20th century. Even armed with the suggestion that the Liar Paradox (**LP**) might somehow be useful as a guide to proving the incompleteness of Peano Arithmetic (**PA**),[1] the level of creativity and philosophical clarity required to actually tie the two concepts together and produce a valid proof is staggering; it certainly should not be controversial to claim that no computational reasoning system can, at present, achieve this sort of feat without significant human assistance.

[1] **G1** of course applies to any axiom system meeting the standard conditions (Turing-decidability, representability, consistency), but we tend to refer to **PA** for economization.

1.1 Automating the Proof of G1

Prior work devoted to producing computational systems able to prove **G1** have yielded systems able to prove this theorem only when the distance between this result and the starting point is quite small. This for example holds for the first (and certainly seminal) foray; i.e., for [Quaife, 1988], as explained in [Bringsjord, 1998], where it's shown that the proof of **G1**, because the set of premises includes an ingenious human-devised encoding scheme, is very easy—to the point of being at the level of proofs requested from students in introductory mathematical logic classes.

Likewise, [Amnon, 1993] is an exact parallel of the human-devised proof given by [Kleene, 1996]. Finally, in much more recent and truly impressive work by [Sieg and Field, 2005], there is a move to natural-deduction formats, which we applaud—but the machine essentially begins its processing at a point exceedingly close to where it needs to end up. As Sieg and Field concede: "As axioms we take for granted the representability and derivability conditions for the central syntactic notions as well as the diagonal lemma for constructing self-referential sentences." If one takes for granted such things, finding a proof of **G1** is effortless for a computing machine.[2] In sum, while a lot of commendable work has been done to build the foundation for our prospective work, the daunting formal and engineering challenge of producing a computational system able to produce **G1** without clever seeding from a human remains entirely unmet.

2 The Analogico-Deductive Approach

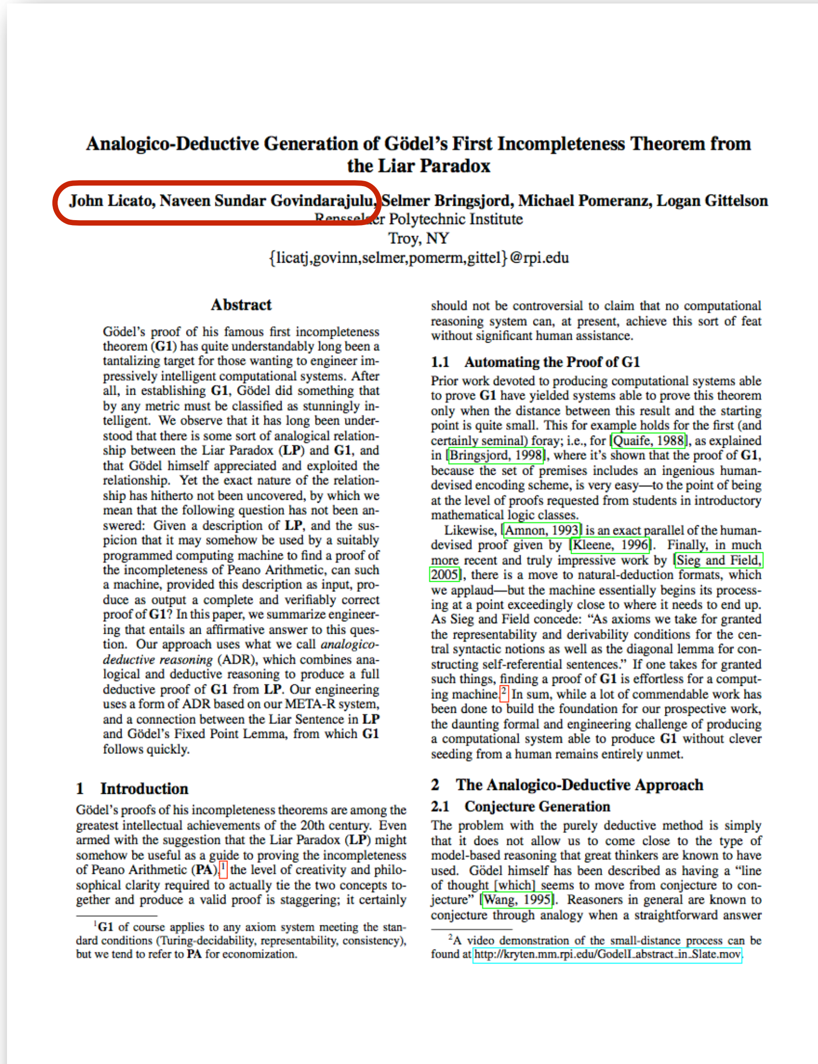
2.1 Conjecture Generation

The problem with the purely deductive method is simply that it does not allow us to come close to the type of model-based reasoning that great thinkers are known to have used. Gödel himself has been described as having a "line of thought [which] seems to move from conjecture to conjecture" [Wang, 1995]. Reasoners in general are known to conjecture through analogy when a straightforward answer

[2] A video demonstration of the small-distance process can be found at http://kryten.mm.rpi.edu/GodelI_abstract_in_Slate.mov.

Licato, J.; Govindarajulu, N.; Bringsjord, S.; Pomeranz, M.; Gittelson, L. 2013. Analogico-Deductive Generation of Godel's First Incompleteness Theorem from the Liar Paradox. In Proceedings of IJCAI 2013. Pdf

“Classic” ADR Result



Analogico-Deductive Generation of Gödel’s First Incompleteness Theorem from the Liar Paradox

John Licato, Naveen Sundar Govindarajulu, Selmer Bringsjord, Michael Pomeranz, Logan Gittelson
Rensselaer Polytechnic Institute
Troy, NY
{licatj,govinn,selmer,pomerm,gittel}@rpi.edu

Abstract

Gödel’s proof of his famous first incompleteness theorem (**G1**) has quite understandably long been a tantalizing target for those wanting to engineer impressively intelligent computational systems. After all, in establishing **G1**, Gödel did something that by any metric must be classified as stunningly intelligent. We observe that it has long been understood that there is some sort of analogical relationship between the Liar Paradox (**LP**) and **G1**, and that Gödel himself appreciated and exploited the relationship. Yet the exact nature of the relationship has hitherto not been uncovered, by which we mean that the following question has not been answered: Given a description of **LP**, and the suspicion that it may somehow be used by a suitably programmed computing machine to find a proof of the incompleteness of Peano Arithmetic, can such a machine, provided this description as input, produce as output a complete and verifiably correct proof of **G1**? In this paper, we summarize engineering that entails an affirmative answer to this question. Our approach uses what we call *analogico-deductive reasoning* (ADR), which combines analogical and deductive reasoning to produce a full deductive proof of **G1** from **LP**. Our engineering uses a form of ADR based on our META-R system, and a connection between the Liar Sentence in **LP** and Gödel’s Fixed Point Lemma, from which **G1** follows quickly.

1 Introduction

Gödel’s proofs of his incompleteness theorems are among the greatest intellectual achievements of the 20th century. Even armed with the suggestion that the Liar Paradox (**LP**) might somehow be useful as a guide to proving the incompleteness of Peano Arithmetic (**PA**),[1] the level of creativity and philosophical clarity required to actually tie the two concepts together and produce a valid proof is staggering; it certainly should not be controversial to claim that no computational reasoning system can, at present, achieve this sort of feat without significant human assistance.

[1] **G1** of course applies to any axiom system meeting the standard conditions (Turing-decidability, representability, consistency), but we tend to refer to **PA** for economization.

1.1 Automating the Proof of G1

Prior work devoted to producing computational systems able to prove **G1** have yielded systems able to prove this theorem only when the distance between this result and the starting point is quite small. This for example holds for the first (and certainly seminal) foray; i.e., for [Quaife, 1988], as explained in [Bringsjord, 1998], where it’s shown that the proof of **G1**, because the set of premises includes an ingenious human-devised encoding scheme, is very easy—to the point of being at the level of proofs requested from students in introductory mathematical logic classes.

Likewise, [Amnon, 1993] is an exact parallel of the human-devised proof given by [Kleene, 1996]. Finally, in much more recent and truly impressive work by [Sieg and Field, 2005], there is a move to natural-deduction formats, which we applaud—but the machine essentially begins its processing at a point exceedingly close to where it needs to end up. As Sieg and Field concede: “As axioms we take for granted the representability and derivability conditions for the central syntactic notions as well as the diagonal lemma for constructing self-referential sentences.” If one takes for granted such things, finding a proof of **G1** is effortless for a computing machine.[2] In sum, while a lot of commendable work has been done to build the foundation for our prospective work, the daunting formal and engineering challenge of producing a computational system able to produce **G1** without clever seeding from a human remains entirely unmet.

2 The Analogico-Deductive Approach

2.1 Conjecture Generation

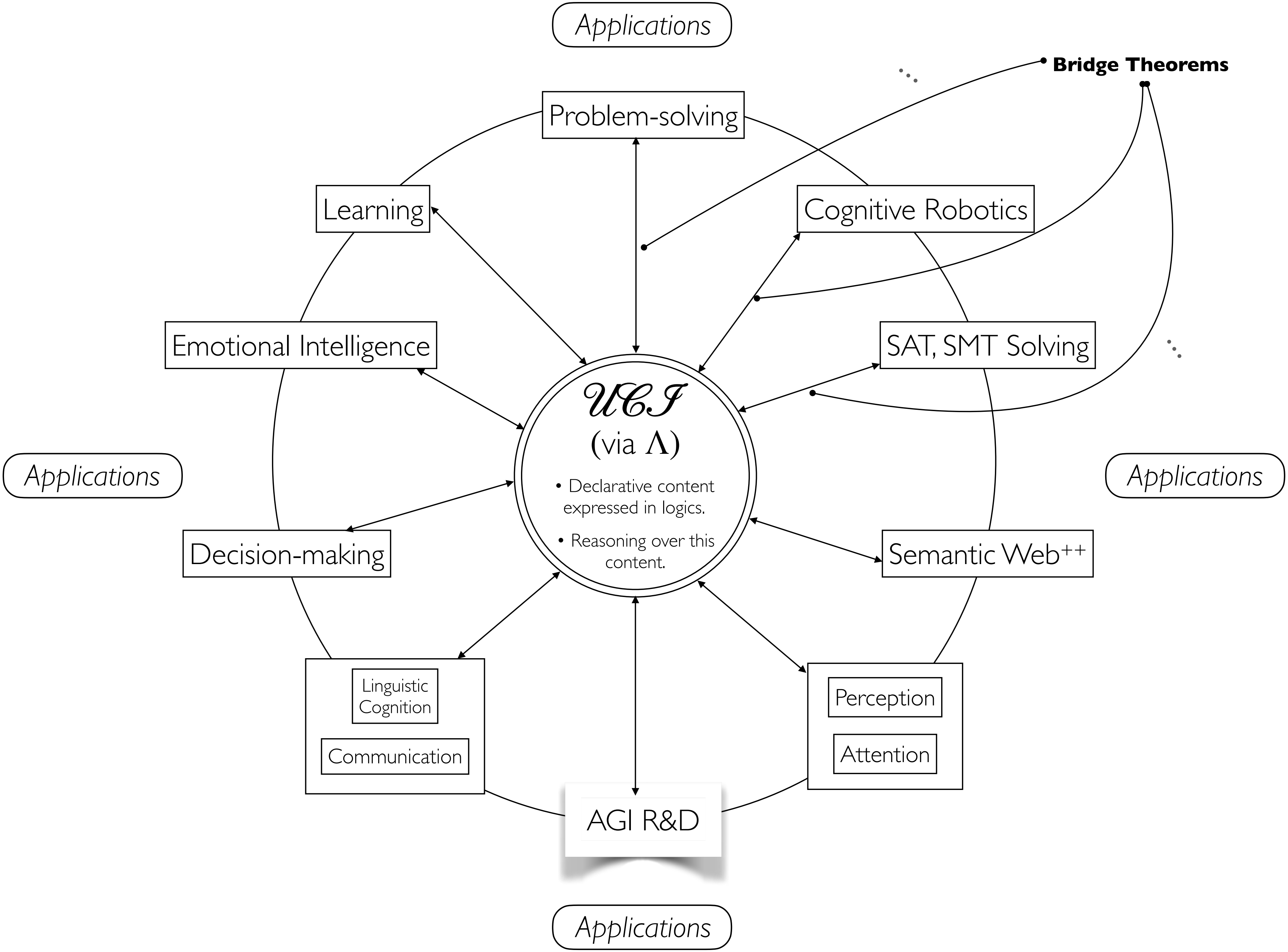
The problem with the purely deductive method is simply that it does not allow us to come close to the type of model-based reasoning that great thinkers are known to have used. Gödel himself has been described as having a “line of thought [which] seems to move from conjecture to conjecture” [Wang, 1995]. Reasoners in general are known to conjecture through analogy when a straightforward answer

[2] A video demonstration of the small-distance process can be found at http://kryten.mm.rpi.edu/GodelI_abstract_in_Slate.mov.

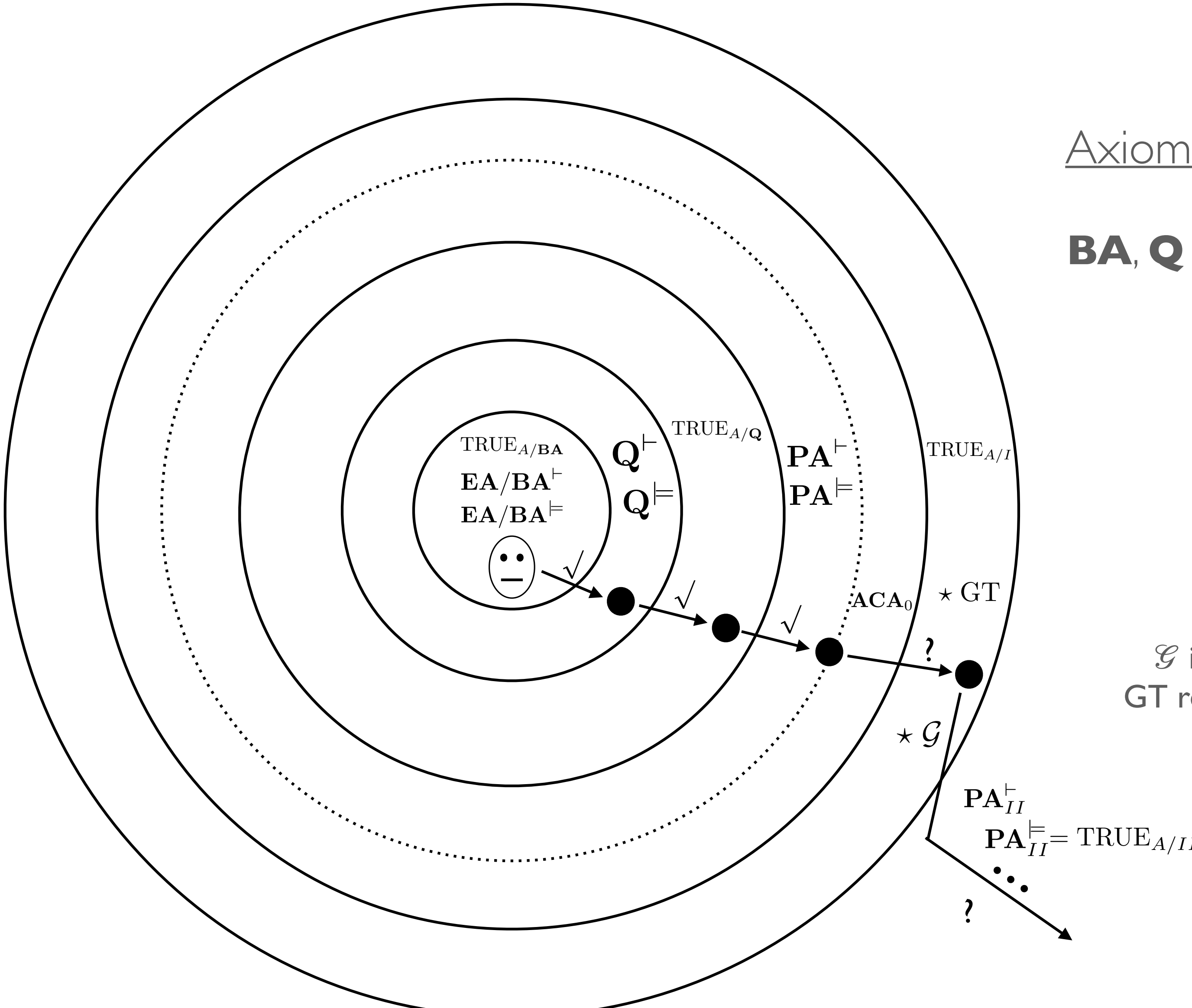
Licato, J.; Govindarajulu, N.; Bringsjord, S.; Pomeranz, M.; Gittelson, L. 2013. Analogico-Deductive Generation of Godel’s First Incompleteness Theorem from the Liar Paradox. In Proceedings of IJCAI 2013. Pdf

4. Logico-mathematically, where is UCI?

. . .


Applications
Bridge Theorems
Problem-solving
Learning
Cognitive Robotics
Emotional Intelligence
SAT, SMT Solving
UCI
(via Λ)
• Declarative content expressed in logics.
• Reasoning over this content.
Applications
Applications
Decision-making
Semantic Web++
Linguistic Cognition
Communication
Perception
Attention
AGI R&D
Applications

(An agent that doesn't know and believe plenty of things on the strength of reasoning isn't intelligent, at all.)

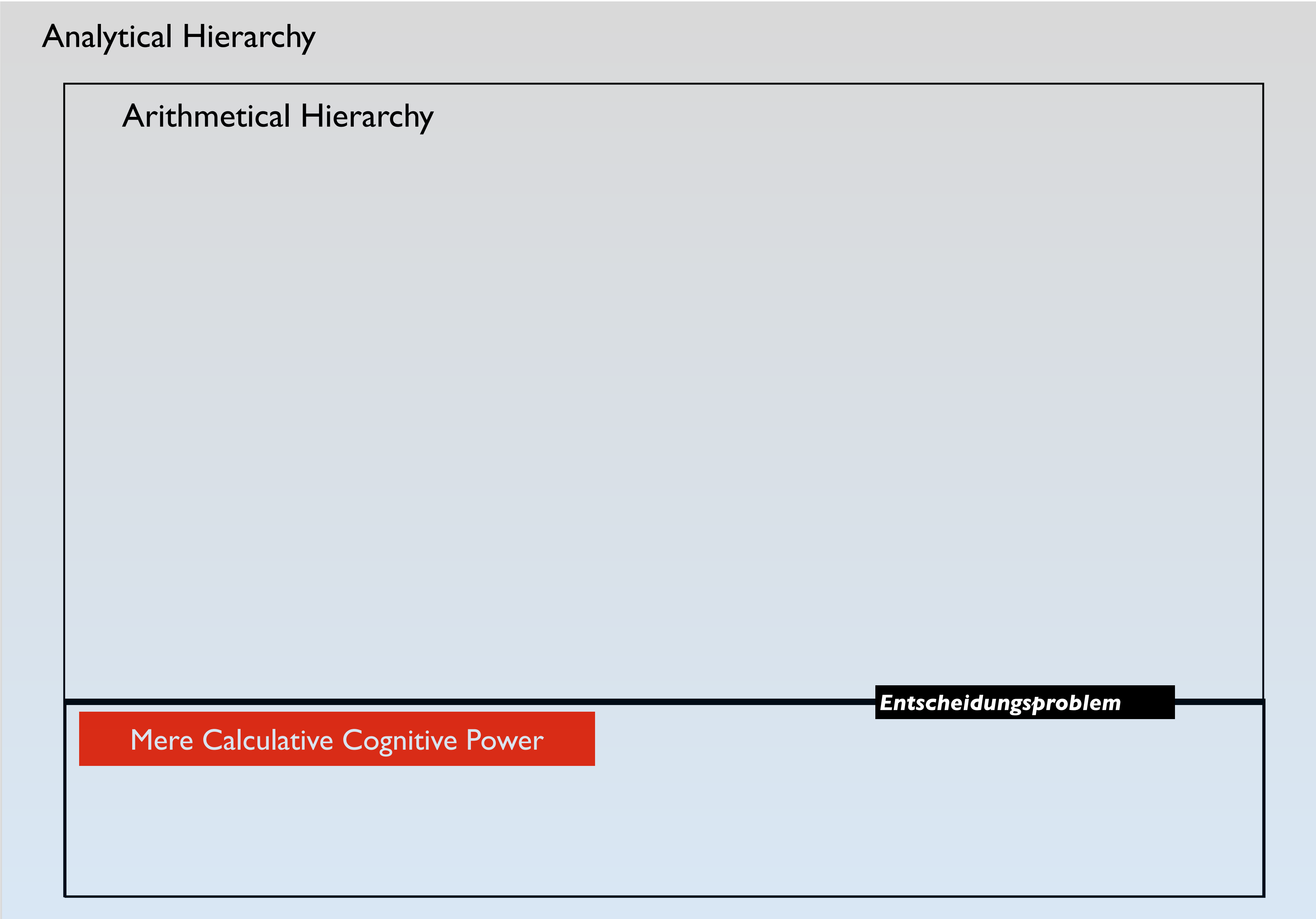


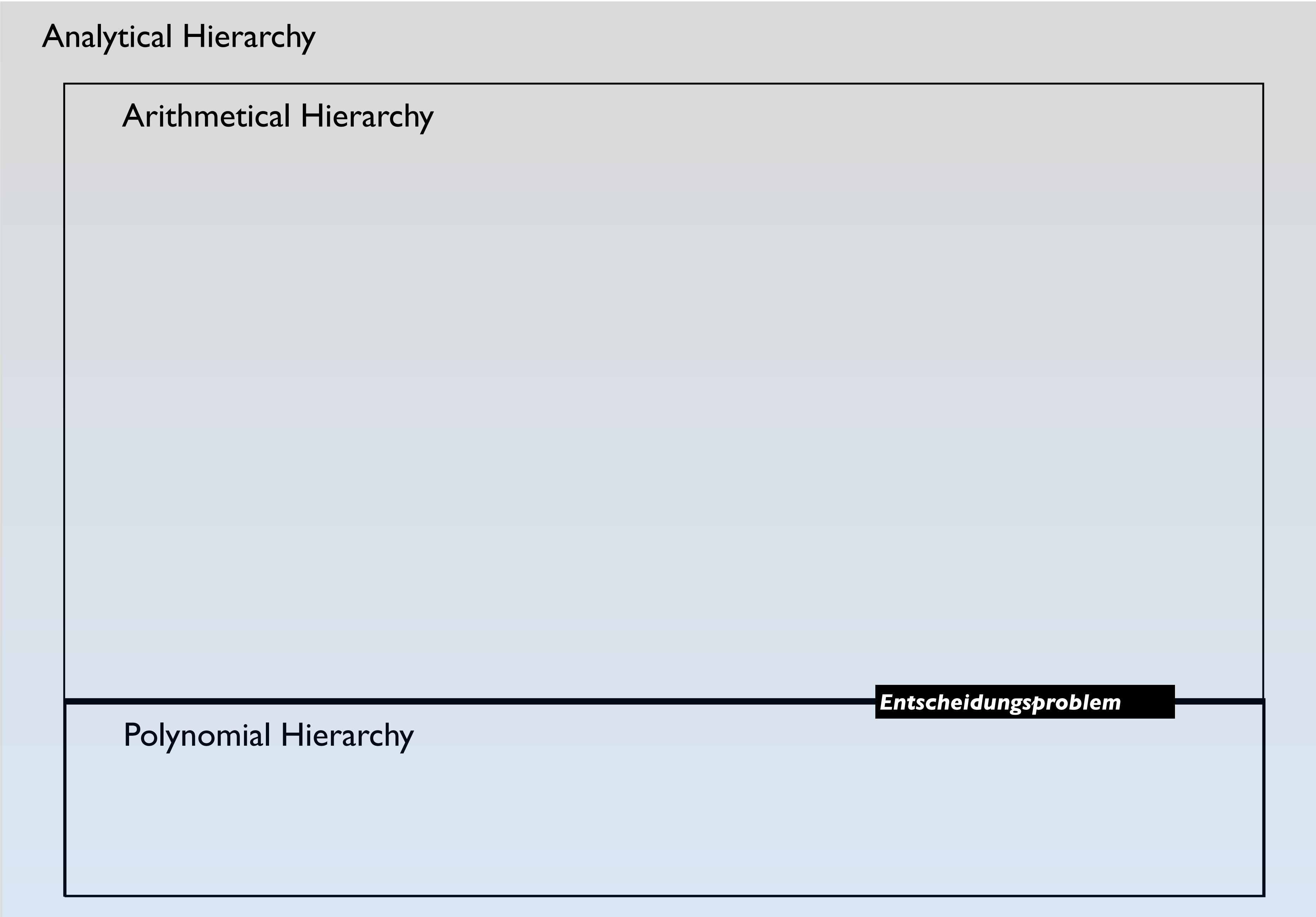
Axiom Systems of Arithmetic:

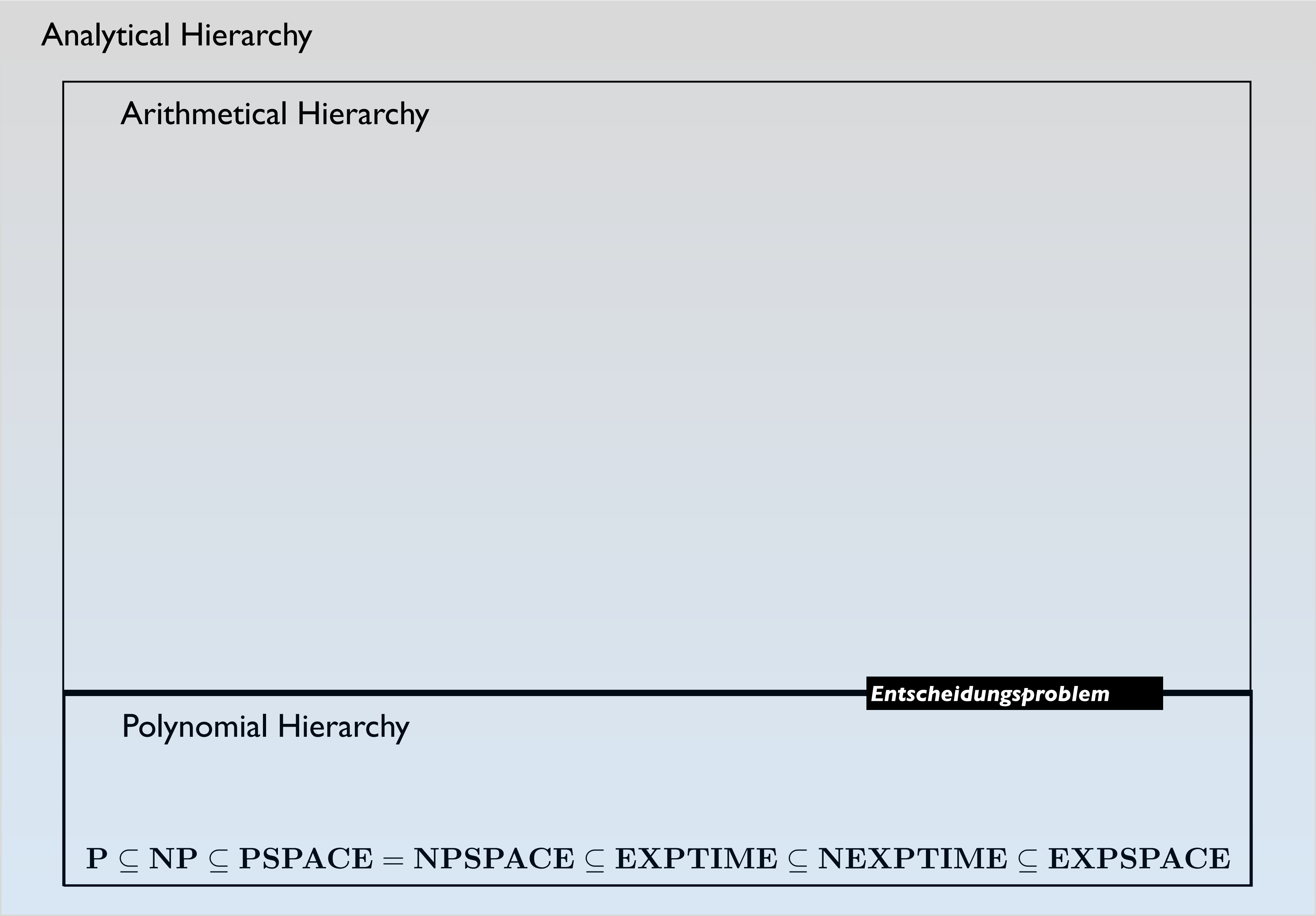
BA, **Q** = **R**, **PA**, $\mathbf{ACA}_0$, $\mathbf{PA}_{II}$, …

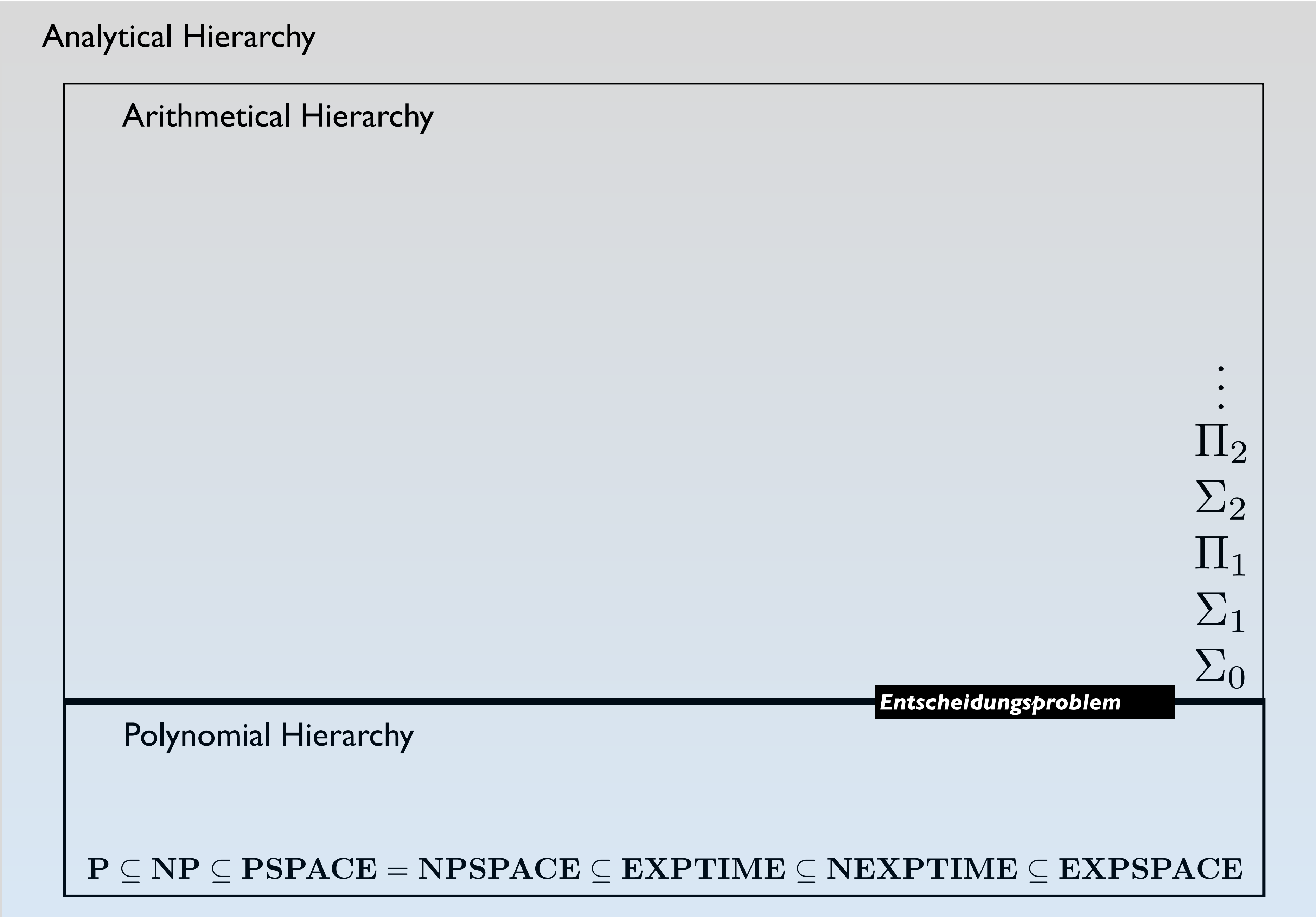
Theorems:

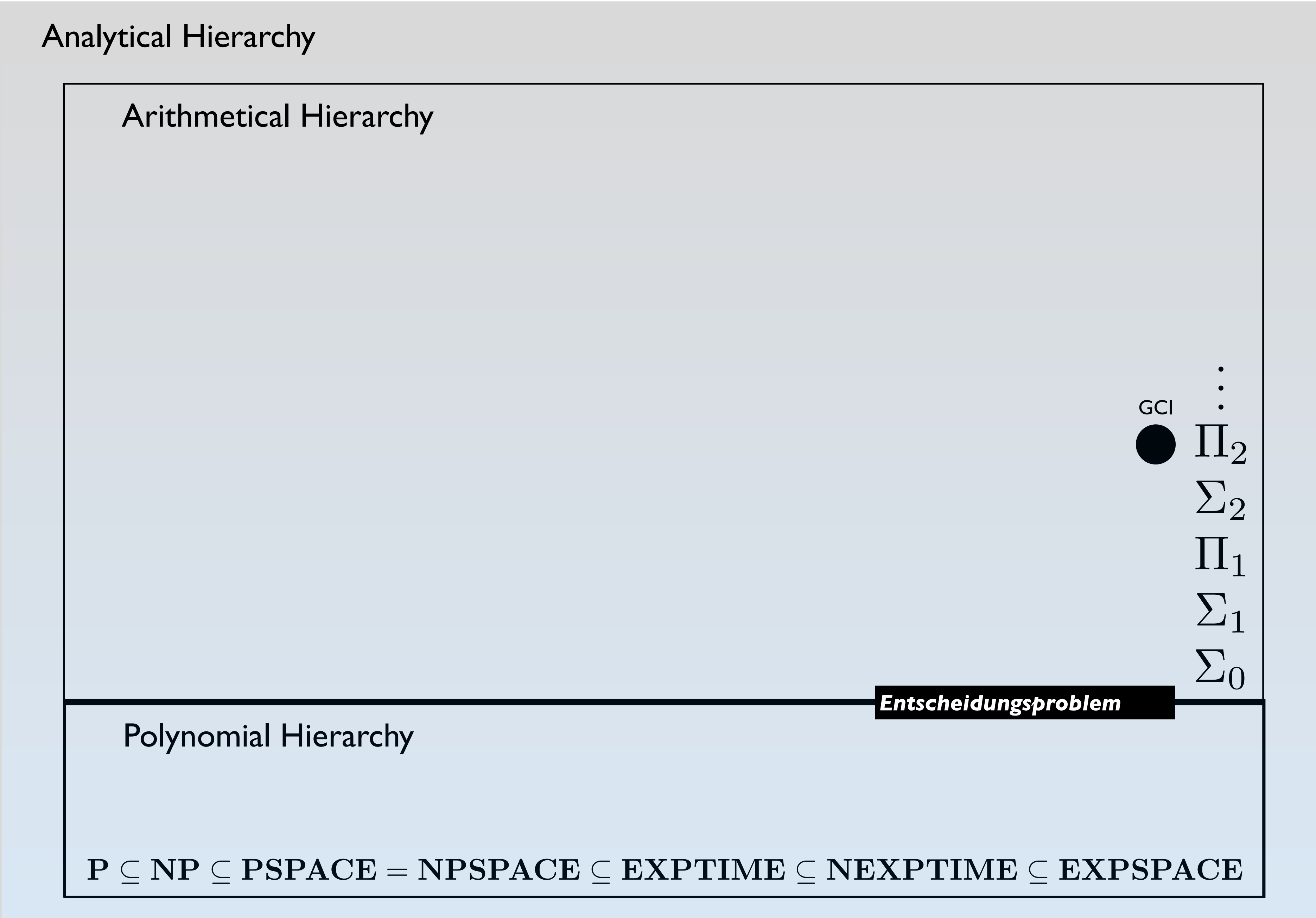
$\mathcal{G}$ is (all of) Gödel's theorems;
GT refers to Goldstein's Theorem.

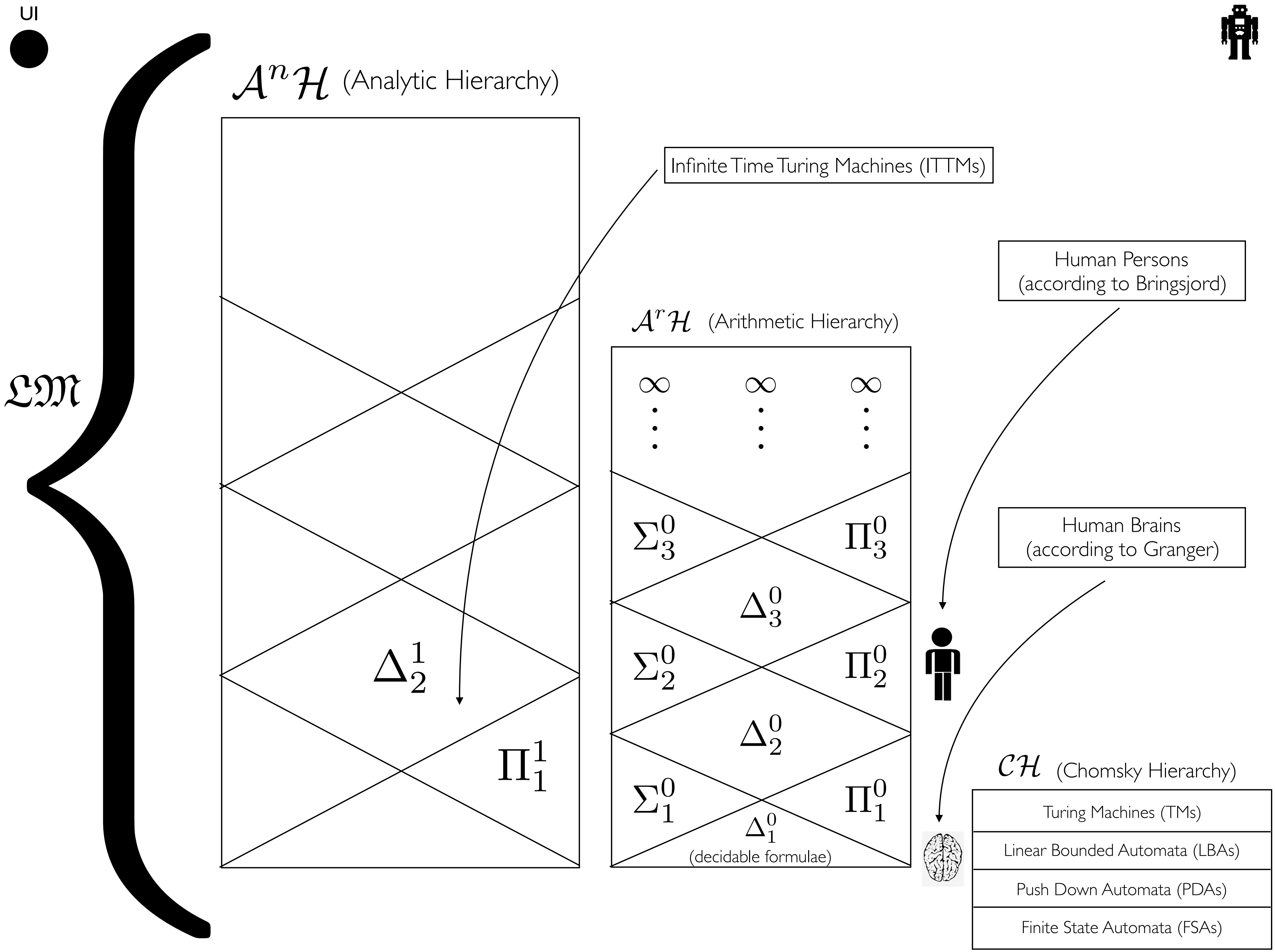

Analytical Hierarchy
Arithmetical Hierarchy
Entscheidungsproblem
Mere Calculative Cognitive Power

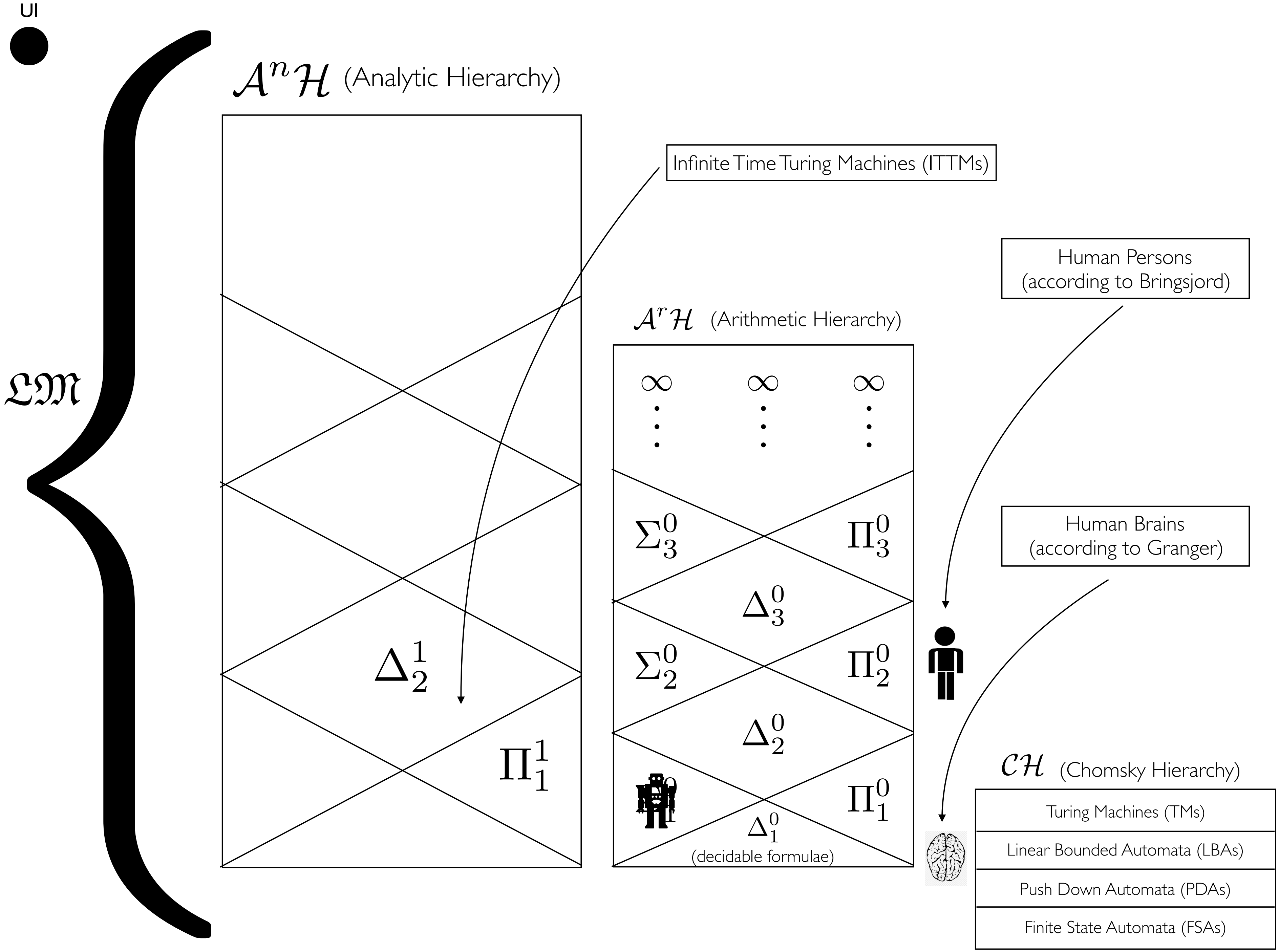

Analytical Hierarchy
Arithmetical Hierarchy
Entscheidungsproblem
Polynomial Hierarchy


Analytical Hierarchy
Arithmetical Hierarchy
Entscheidungsproblem
Polynomial Hierarchy
P ⊆ NP ⊆ PSPACE = NPSPACE ⊆ EXPTIME ⊆ NEXPTIME ⊆ EXPSPACE

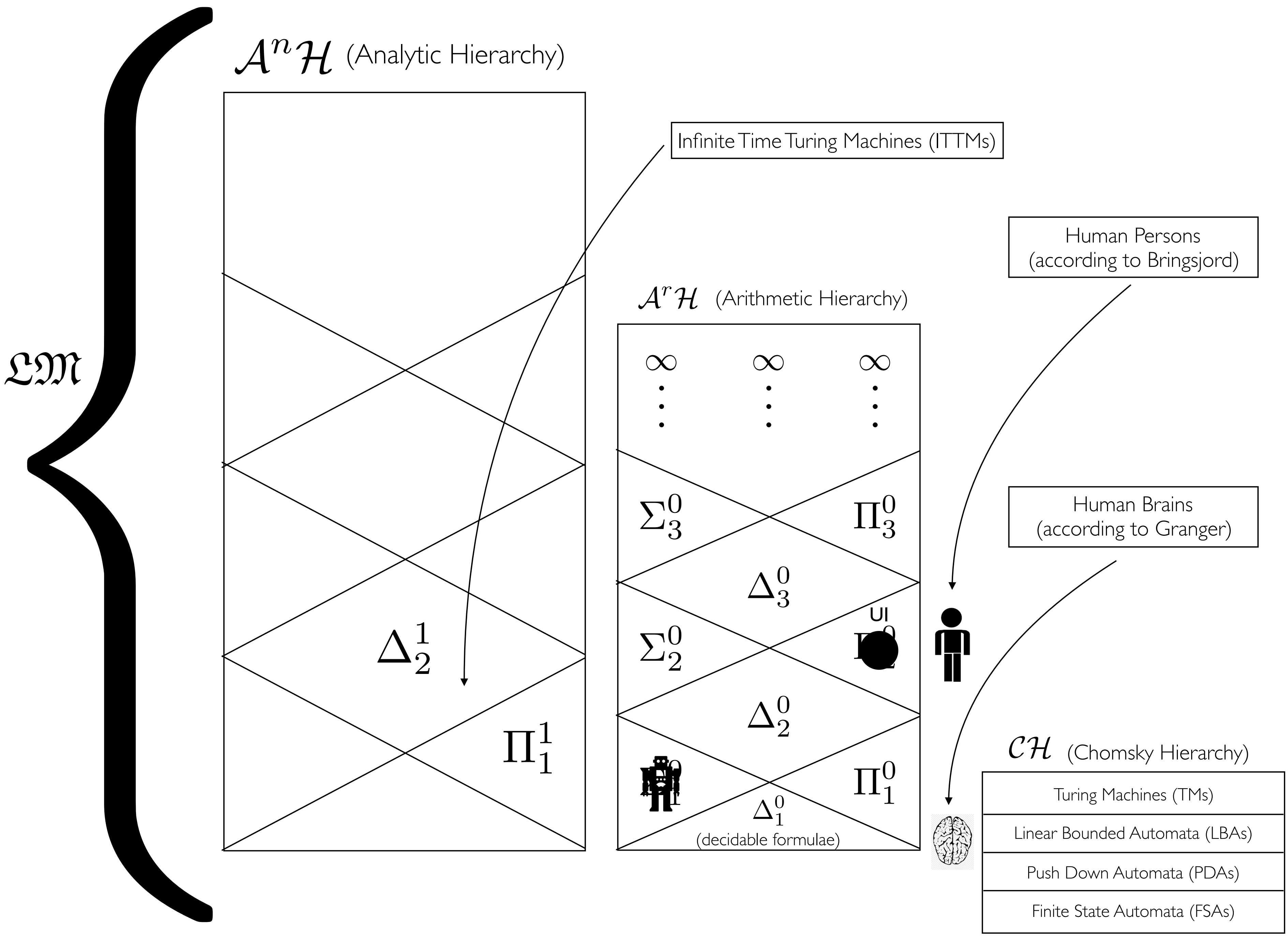

Analytical Hierarchy
Arithmetical Hierarchy
$\vdots$
Π_2
Σ_2
Π_1
Σ_1
Σ_0
Entscheidungsproblem
Polynomial Hierarchy
$\mathbf{P} \subseteq \mathbf{NP} \subseteq \mathbf{PSPACE} = \mathbf{NPSPACE} \subseteq \mathbf{EXPTIME} \subseteq \mathbf{NEXPTIME} \subseteq \mathbf{EXPSPACE}$


Analytical Hierarchy
Arithmetical Hierarchy
GCI
$\vdots$
Π_2
Σ_2
Π_1
Σ_1
Σ_0
Entscheidungsproblem
Polynomial Hierarchy
$\mathbf{P} \subseteq \mathbf{NP} \subseteq \mathbf{PSPACE} = \mathbf{NPSPACE} \subseteq \mathbf{EXPTIME} \subseteq \mathbf{NEXPTIME} \subseteq \mathbf{EXPSPACE}$

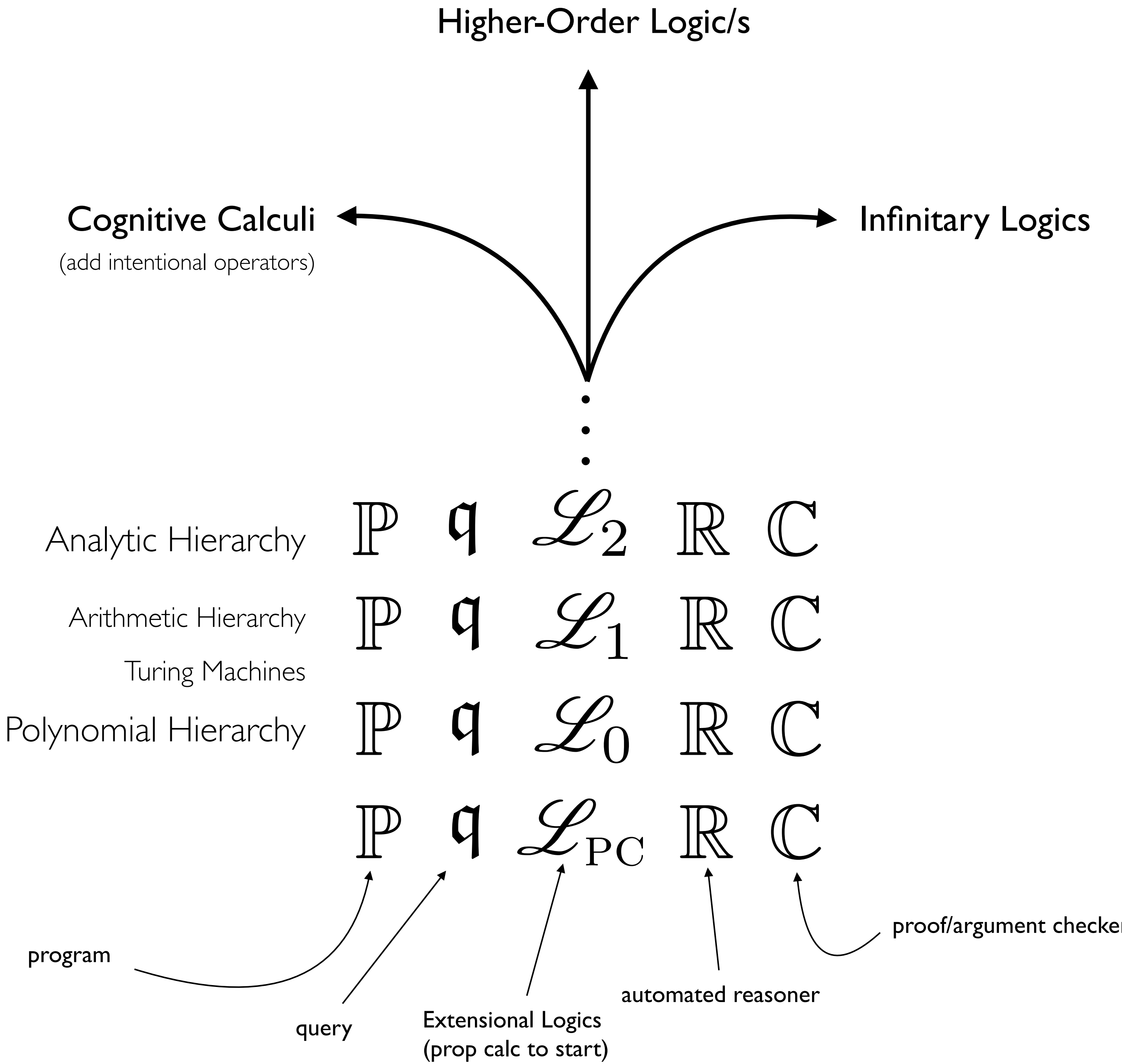

UI
$\mathcal{A}^n\mathcal{H}$ (Analytic Hierarchy)
Infinite Time Turing Machines (ITTMs)
Human Persons
(according to Bringsjord)
$\mathcal{A}^r\mathcal{H}$ (Arithmetic Hierarchy)
$\mathfrak{LM}$
∞
∞
∞
Σ_3^0
Π_3^0
Human Brains
(according to Granger)
Δ_3^0
Δ_2^1
Σ_2^0
Π_2^0
Δ_2^0
$\mathcal{CH}$ (Chomsky Hierarchy)
Π_1^1
Σ_1^0
Π_1^0
Δ_1^0
(decidable formulae)
Turing Machines (TMs)
Linear Bounded Automata (LBAs)
Push Down Automata (PDAs)
Finite State Automata (FSAs)

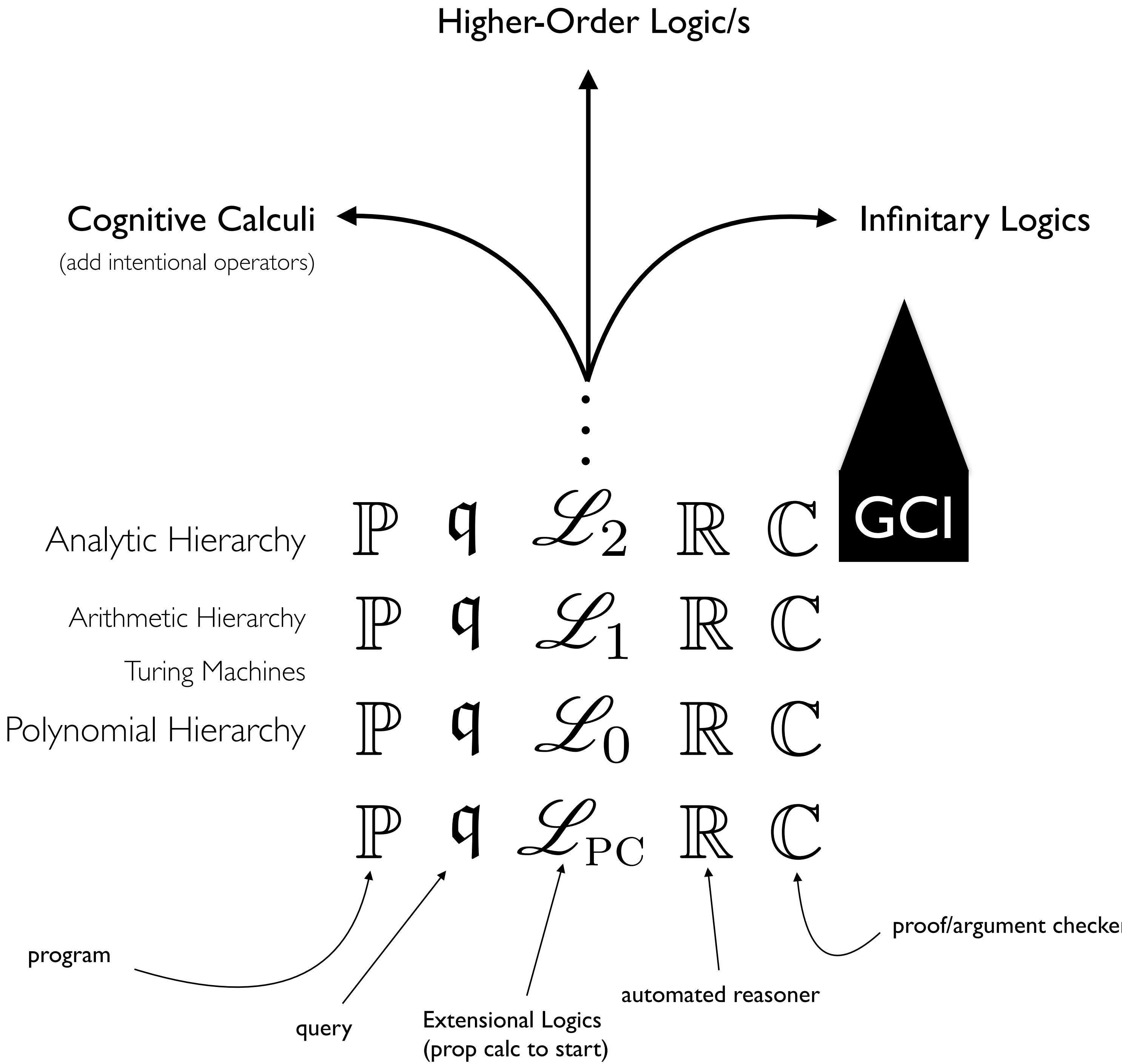


UI
$\mathfrak{LM}$
$\mathcal{A}^n\mathcal{H}$ (Analytic Hierarchy)
Infinite Time Turing Machines (ITTMs)
Δ^1_2
Π^1_1
$\mathcal{A}^r\mathcal{H}$ (Arithmetic Hierarchy)
∞ ∞ ∞
Σ^0_3 Π^0_3
Δ^0_3
Σ^0_2 Π^0_2
Δ^0_2
Π^0_1
Δ^0_1
(decidable formulae)
Human Persons
(according to Bringsjord)
Human Brains
(according to Granger)
$\mathcal{CH}$ (Chomsky Hierarchy)
Turing Machines (TMs)
Linear Bounded Automata (LBAs)
Push Down Automata (PDAs)
Finite State Automata (FSAs)

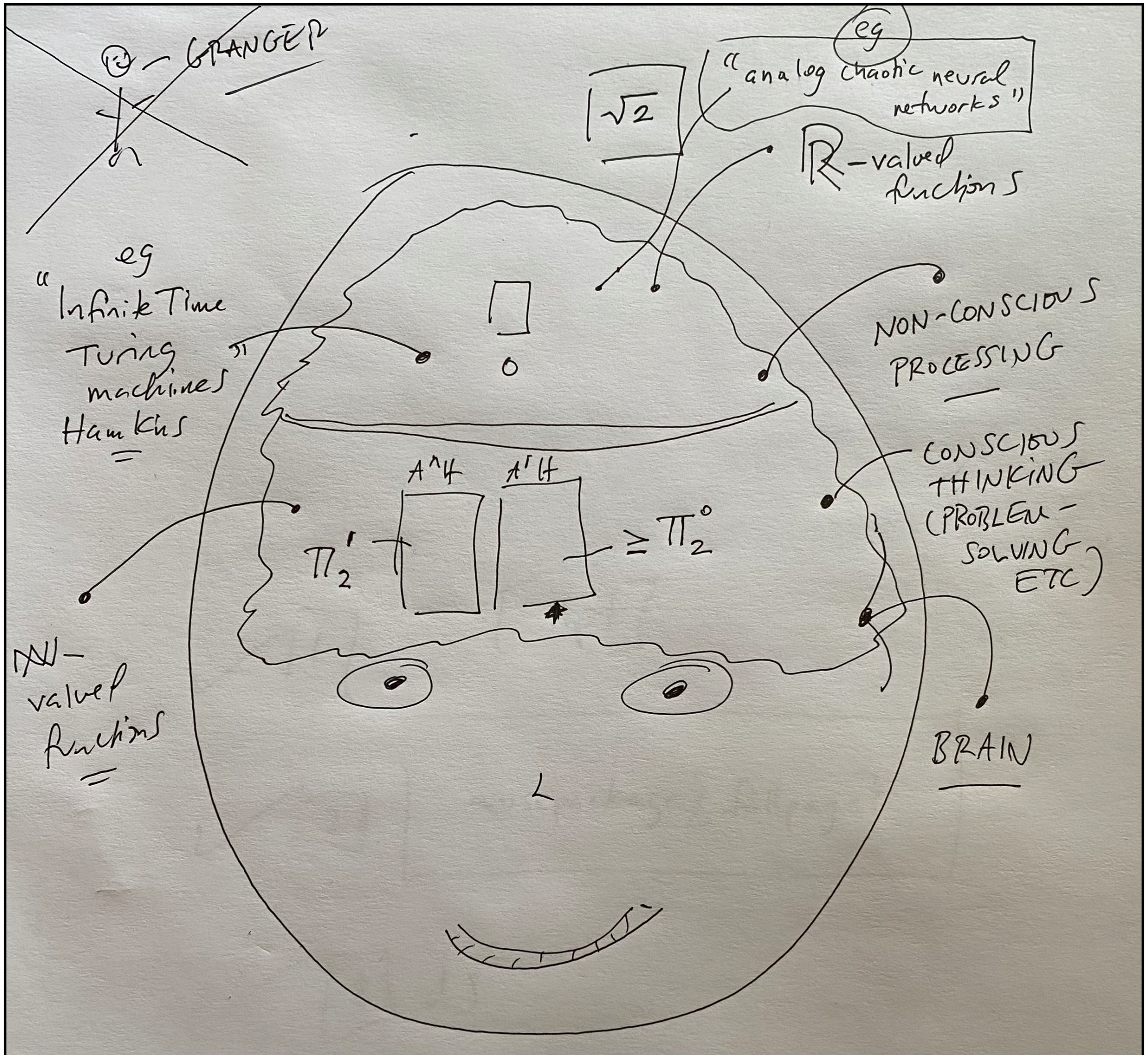


$\mathfrak{LM}$
$\mathcal{A}^n\mathcal{H}$ (Analytic Hierarchy)
Δ^1_2
Π^1_1
Infinite Time Turing Machines (ITTMs)
$\mathcal{A}^r\mathcal{H}$ (Arithmetic Hierarchy)
∞ ∞ ∞
Σ^0_3
Π^0_3
Δ^0_3
UI
Σ^0_2
Δ^0_2
Π^0_1
Δ^0_1
(decidable formulae)
Human Persons
(according to Bringsjord)
Human Brains
(according to Granger)
$\mathcal{CH}$ (Chomsky Hierarchy)
Turing Machines (TMs)
Linear Bounded Automata (LBAs)
Push Down Automata (PDAs)
Finite State Automata (FSAs)

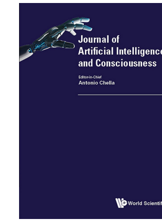

Higher-Order Logic/s
Cognitive Calculi
(add intentional operators)
Infinitary Logics
Analytic Hierarchy
Arithmetic Hierarchy
Turing Machines
Polynomial Hierarchy
$\mathbb{P}$ $\mathfrak{q}$ $\mathcal{L}_2$ $\mathbb{R}$ $\mathbb{C}$
$\mathbb{P}$ $\mathfrak{q}$ $\mathcal{L}_1$ $\mathbb{R}$ $\mathbb{C}$
$\mathbb{P}$ $\mathfrak{q}$ $\mathcal{L}_0$ $\mathbb{R}$ $\mathbb{C}$
$\mathbb{P}$ $\mathfrak{q}$ $\mathcal{L}_{\mathrm{PC}}$ $\mathbb{R}$ $\mathbb{C}$
program
query
Extensional Logics
(prop calc to start)
automated reasoner
proof/argument checker


Higher-Order Logic/s
Cognitive Calculi
(add intentional operators)
Infinitary Logics
GCI
Analytic Hierarchy
Arithmetic Hierarchy
Turing Machines
Polynomial Hierarchy
$\mathbb{P}$ $\mathfrak{q}$ $\mathscr{L}_2$ $\mathbb{R}$ $\mathbb{C}$
$\mathbb{P}$ $\mathfrak{q}$ $\mathscr{L}_1$ $\mathbb{R}$ $\mathbb{C}$
$\mathbb{P}$ $\mathfrak{q}$ $\mathscr{L}_0$ $\mathbb{R}$ $\mathbb{C}$
$\mathbb{P}$ $\mathfrak{q}$ $\mathscr{L}_{\mathrm{PC}}$ $\mathbb{R}$ $\mathbb{C}$
program
query
Extensional Logics
(prop calc to start)
automated reasoner
proof/argument checker

Picture of HI, *Contra* Granger



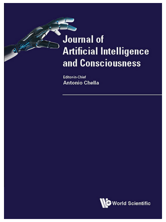
Journal of Artificial Intelligence and Consciousness | Vol. 07, No. 02, pp. 155-181 (2020)



The Theory of Cognitive Consciousness, and Λ (Lambda)

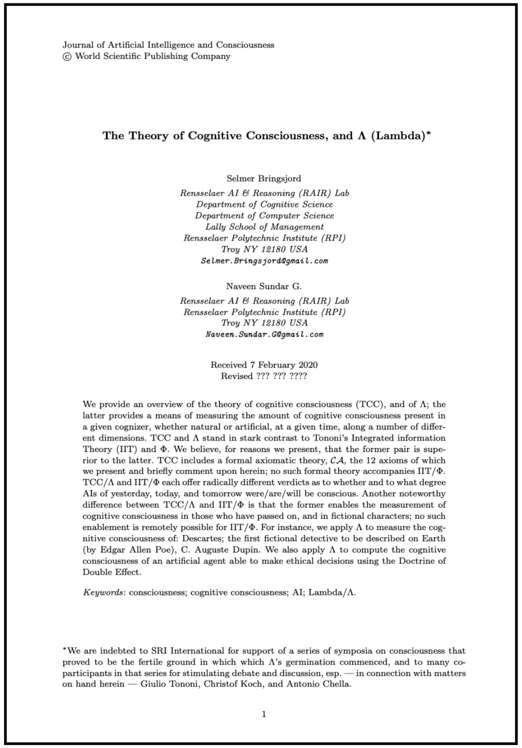
Selmer Bringsjord and G. Naveen Sundar

Journal of Artificial Intelligence and Consciousness | Vol. 07, No. 02, pp. 155-181 (2020)



The Theory of Cognitive Consciousness, and Λ (Lambda)

Selmer Bringsjord and G. Naveen Sundar



Journal of Artificial Intelligence and Consciousness

The Theory of Cognitive Consciousness, and Λ (Lambda)*

Selmer Bringsjord
Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science
Department of Computer Science
Lally School of Management
Rensselaer Polytechnic Institute (RPI)
Troy NY 12180 USA
Selmer.Bringsjord@gmail.com

Naveen Sundar G.
Rensselaer AI & Reasoning (RAIR) Lab
Rensselaer Polytechnic Institute (RPI)
Troy NY 12180 USA
Naveen.Sundar.G@gmail.com

Received 7 February 2020
Revised ??? ??? ????

We provide an overview of the theory of cognitive consciousness (TCC), and of Λ; the latter provides a means of measuring the amount of cognitive consciousness present in a given cognizer, whether natural or artificial, at a given time, along a number of different dimensions. TCC and Λ stand in stark contrast to Tononi's Integrated information Theory (IIT) and Φ. We believe, for reasons we present, that the former pair is superior to the latter. TCC includes a formal axiomatic theory, $\mathcal{CA}$, the 12 axioms of which we present and briefly comment upon herein; no such formal theory accompanies IIT/Φ. TCC/Λ and IIT/Φ each offer radically different verdicts as to whether and to what degree AIs of yesterday, today, and tomorrow were/are/will be conscious. Another noteworthy difference between TCC/Λ and IIT/Φ is that the former enables the measurement of cognitive consciousness in those who have passed on, and in fictional characters; no such enablement is remotely possible for IIT/Φ. For instance, we apply Λ to measure the cognitive consciousness of: Descartes; the first fictional detective to be described on Earth (by Edgar Allen Poe), C. Auguste Dupin. We also apply Λ to compute the cognitive consciousness of an artificial agent able to make ethical decisions using the Doctrine of Double Effect.

Keywords: consciousness; cognitive consciousness; AI; Lambda/Λ.

*We are indebted to SRI International for support of a series of symposia on consciousness that proved to be the fertile ground in which which Λ's germination commenced, and to many co-participants in that series for stimulating debate and discussion, esp. — in connection with matters on hand herein — Giulio Tononi, Christof Koch, and Antonio Chella.

Journal of Artificial Intelligence and Consciousness | Vol. 07, No. 02, pp. 155-181 (2020)

The Theory of Cognitive Consciousness, and Λ (Lambda)

Extending Measures from $\mathcal{L}^0$ to $\mathcal{L}$

$$\mu_\omega(\phi) = \begin{cases} \mu(\phi) & \text{if } \phi \in \mathcal{L}^0 \\ \max_\psi \mu_\omega(\psi) + 1 & \text{if } \phi \equiv \omega_i\,[a_1, t_1, \ldots \psi \ldots] \end{cases}$$

For example, let μ count the number of predicate symbols in a formula.

Example

$$\mu\,(Happy\,(john)) = 1$$
$$\mu_\omega\big(Happy\,(john)\big) = 1$$
$$\mu_\omega\Big(\mathbf{B}\big(mary, t_2, Happy(john)\big)\Big) = 2$$

For any agent a, we want to look at the new complexity the agent introduces that is above any input complexity. For this, we introduce $\Delta : 2^{\mathcal{L}} \times 2^{\mathcal{L}} \rightarrow 2^{\mathcal{L}}$ operator that computes differences between two sets of formulae. This can be simply the set-difference operator. For convenience, let $\omega_j[\Gamma]$ denote the subset of formulae with operators ω_j in Γ:

$$\omega_j[\Gamma] = \{\phi \mid \ \phi \equiv \omega_j[\ldots] \text{ and } \phi \in \Gamma \text{ or } \phi \text{ a subformula } \in \Gamma\}$$

Given a set of measures $\{\mu^0, \ldots, \mu^N\}$ and a set of modal (or cognitive) operators $\{\omega_0, \ldots, \omega_M\}$, we define Λ as a function mapping an agent at a time point to a matrix $\mathbb{N}^{M\times N}$:

$$\Lambda : A \times T \rightarrow \mathbb{N}^{M\times N}$$

Definition of Λ

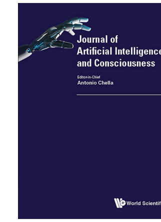
$$\Lambda(a,t)_{i,j} = \max_{\phi}\Big\{\mu^i(\phi) \mid \phi \in \Delta\Big(\omega_j\big[o(a,t)\big], \omega_j\big[i(a,t)\big]\Big)\Big\}$$

Example 2

Let us consider two modal operators $\{\mathbf{B}, \mathbf{D}\}$ and the following base measures μ^0 which measures quantificational complexity via Σ or Π measures, μ^1 which counts the total number of predicate symbols (not a count of unique predicate symbols), and μ^2 which counts the number of distinct time expressions. This gives $\Lambda : A \times T \rightarrow \mathbb{N}^{2\times 3}$. At some timepoint t, let an agent a have the following $\Delta(o(a,t), i(a,t)) = \{\mathbf{B}(\phi_1), \mathbf{D}(\phi_2)\}$

Journal of Artificial Intelligence and Consciousness | Vol. 07, No. 02, pp. 155-181 (2020)

The Theory of Cognitive Consciousness, and Λ (Lambda)



Extending Measures from $\mathcal{L}^0$ to $\mathcal{L}$

$$\mu_\omega(\phi) = \begin{cases} \mu(\phi) & \text{if } \phi \in \mathcal{L}^0 \\ \max_\psi \mu_\omega(\psi) + 1 & \text{if } \phi \equiv \omega_i [a_1, t_1, \ldots \psi \ldots] \end{cases}$$

For example, let μ count the number of predicate symbols in a formula.

Example

$$\mu\,(Happy\,(john)) = 1$$
$$\mu_\omega\big(Happy\,(john)\big) = 1$$
$$\mu_\omega\Big(\mathbf{B}\big(mary, t_2, Happy(john)\big)\Big) = 2$$

For any agent a, we want to look at the new complexity the agent introduces that is above any input complexity. For this, we introduce $\Delta : 2^{\mathcal{L}} \times 2^{\mathcal{L}} \rightarrow 2^{\mathcal{L}}$ operator that computes differences between two sets of formulae. This can be simply the set-difference operator. For convenience, let $\omega_j[\Gamma]$ denote the subset of formulae with operators ω_j in Γ:

$$\omega_j[\Gamma] = \{\phi \mid \ \phi \equiv \omega_j[\ldots] \text{ and } \phi \in \Gamma \text{ or } \phi \text{ a subformula } \in \Gamma\}$$

Given a set of measures $\{\mu^0, \ldots, \mu^N\}$ and a set of modal (or cognitive) operators $\{\omega_0, \ldots, \omega_M\}$, we define Λ as a function mapping an agent at a time point to a matrix $\mathbb{N}^{M \times N}$:

$$\Lambda : A \times T \rightarrow \mathbb{N}^{M \times N}$$

Definition of Λ

$$\Lambda(a,t)_{i,j} = \max_\phi \Big\{\mu^i(\phi) \mid \phi \in \Delta\Big(\omega_j\big[o(a,t)\big], \omega_j\big[i(a,t)\big]\Big)\Big\}$$

Example 2

Let us consider two modal operators $\{\mathbf{B}, \mathbf{D}\}$ and the following base measures μ^0 which measures quantificational complexity via Σ or Π measures, μ^1 which counts the total number of predicate symbols (not a count of unique predicate symbols), and μ^2 which counts the number of distinct time expressions. This gives $\Lambda : A \times T \rightarrow \mathbb{N}^{2\times 3}$. At some timepoint t, let an agent a have the following $\Delta(o(a,t), i(a,t)) = \{\mathbf{B}(\phi_1), \mathbf{D}(\phi_2)\}$

$$\phi_1 \equiv \neg\forall a : Happy(a,t); \qquad \phi_2 \equiv \forall b : \neg Hungry(b,t) \rightarrow Happy(b,t)$$

Applying the measures:

$$\mu^o(\phi_1) = 1, \mu^1(\phi_1) = 1; \mu^2(\phi_1) = 1$$
$$\mu^o(\phi_2) = 1; \mu^1(\phi_2) = 2; \mu^2(\phi_2) = 1$$

Giving us:

$$\Lambda(a,t) = \begin{pmatrix} 1\,1\,1 \\ 1\,2\,1 \end{pmatrix}$$

6.1. *Some Distinctive Properties of* Λ *(vs.* Φ*)*

Here are some properties of the Λ framework of potential interest to our readers:

Non-Binary Whereas Φ is such that an agent either is or is not (P-) conscious, cognitive consciousness as measured by Λ admits of a fine-grained range of the *degree* of cognitive consciousness.

Zero Λ for Some Animals and Machines Animals such as insects, and computing machines that are end-to-end statistical/connectionist "ML," have zero Λ, and hence cannot be cognitively conscious. In contrast, as emphasized to Bringsjord in personal conversation,[6] Φ says that even lower animals are conscious.

Human-Nonhuman Discontinuity Explained by Λ From the computational/AI point of view, cognitive scientists have taken note of a severe discontinuity between *H. sapiens sapiens* and other biological creatures on Earth [Penn *et al.*, 2008], and the sudden and large jump in level of Λ from (say) chimpanzees and dolphins to humans is in line with this observation. It's for instance doubtful that any nonhuman animals are capable of reaching third-order belief; hence $\Lambda[\mathbf{B}, 0] = n$, where $n \geq 3$, for any nonhuman animal, is impossible. In stark contrast, each of us believes that you, the reader, believe that we believe that San Francisco is located in California.

Human-Human Discontinuity Explained by Λ A given neurobiologically normal human, over the course of his or her lifetime, has very different cognitive capacity. E.g., it's well-known that such a human, before the age of four or five, is highly unlikely to be able to solve what has become known as the *false-belief task* (or sometimes the *sally-anne task*), which we denote by 'FBT.' From the point of view of Λ, the explanation is simply that an agent with insufficiently high cognitive consciousness is incapable of solving such a task; specifically, solving FBT requires an agent to have

[6] With Tononi and C. Koch, SRI T&C Series.

$\mathcal{CA}$: 11 Axioms (Initially)

Plan

P2B

K2B

Intro

Incorr

Ess

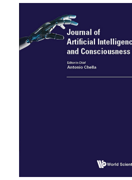
¬CompE

Irr

Free

CCaus

TheI



$\mathcal{CA}$: 11 Axioms (Initially)

Plan

P2B

K2B $\forall a[\mathbf{K}_a \phi \rightarrow (\mathbf{B}_a \phi \wedge \mathbf{B}_a \exists \Phi \exists \alpha (\Phi \leadsto_{\alpha/\pi} \phi)]$

Intro

Incorr

Ess

¬CompE

Irr

Free

CCaus

TheI

$\mathcal{CA}$: 11 Axioms (Initially)

Plan

P2B

$\mu\mathcal{DCEC}_3^*$ K2B $\forall a[\mathbf{K}_a\phi \rightarrow (\mathbf{B}_a\phi \wedge \mathbf{B}_a\exists\Phi\exists\alpha(\Phi \rightsquigarrow_{\alpha/\pi} \phi)]$

Intro

Incorr

Ess

¬CompE

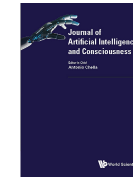
Irr

Free

CCaus

TheI

$\mathcal{CA}$: 11 Axioms (Initially)



Plan

P2B

K2B $\forall a[\mathbf{K}_a\phi \rightarrow (\mathbf{B}_a\phi \wedge \mathbf{B}_a\exists\Phi\exists\alpha(\Phi \rightsquigarrow_{\alpha/\pi} \phi)]$

Intro

Incorr

Ess

¬CompE

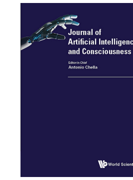
Irr

Free

CCaus

TheI

$\mathcal{CA}$: 11 Axioms (Initially)



Plan

P2B

K2B $\forall a[\mathbf{K}_a\phi \rightarrow (\mathbf{B}_a\phi \land \mathbf{B}_a\exists\Phi\exists\alpha(\Phi \leadsto_{\alpha/\pi} \phi)]$

Intro

Incorr $\forall a \forall t \forall F[(F \mathit{is\ contingent} \ \land F \in C'') \rightarrow (\Box\mathbf{B}(a,t,Fa) \rightarrow Fa)]$

Ess

¬CompE

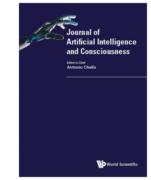
Irr

Free

CCaus

TheI

$\mathcal{CA}$: 11 Axioms (Initially)



Plan

P2B

K2B $\forall a[\mathbf{K}_a\phi \rightarrow (\mathbf{B}_a\phi \wedge \mathbf{B}_a\exists\Phi\exists\alpha(\Phi \rightsquigarrow_{\alpha/\pi} \phi))]$

Intro

Incorr $\forall a\forall t\forall F[(F\, is\ contingent \ \wedge F \in C'') \rightarrow (\Box\mathbf{B}(a,t,Fa) \rightarrow Fa)]$

Ess

¬CompE

Irr

Free

CCaus $\mathbf{C}$ $\mathcal{EC}$

TheI

$\mathcal{CA}$: 11 Axioms (Initially)

Plan

P2B

K2B $\forall a[\mathbf{K}_a \phi \rightarrow (\mathbf{B}_a \phi \wedge \mathbf{B}_a \exists \Phi \exists \alpha (\Phi \rightsquigarrow_{\alpha/\pi} \phi)]$

Intro

Incorr $\forall a \forall t \forall F[(F \mathit{is\ contingent} \ \wedge F \in C'') \rightarrow (\Box \mathbf{B}(a,t,Fa) \rightarrow Fa)]$

Ess

¬CompE

Irr

Free

CCaus $\mathbf{C}$ $\mathcal{EC}$

TheI

$[A_1]$ $\mathbf{C}(\forall f,t \ . \ \mathit{initially}(f) \wedge \neg \mathit{clipped}(0,f,t) \Rightarrow \mathit{holds}(f,t))$

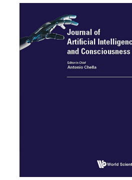
$[A_2]$ $\mathbf{C}(\forall e,f,t_1,t_2 \ . \ \mathit{happens}(e,t_1) \wedge \mathit{initiates}(e,f,t_1) \wedge t_1 < t_2 \wedge \neg \mathit{clipped}(t_1,f,t_2) \Rightarrow \mathit{holds}(f,t_2))$

$[A_3]$ $\mathbf{C}(\forall t_1,f,t_2 \ . \ \mathit{clipped}(t_1,f,t_2) \Leftrightarrow [\exists e,t \ . \ \mathit{happens}(e,t) \wedge t_1 < t < t_2 \wedge \mathit{terminates}(e,f,t)])$

$[A_4]$ $\mathbf{C}(\forall a,d,t \ . \ \mathit{happens}(\mathit{action}(a,d),t) \Rightarrow \mathbf{K}(a, \mathit{happens}(\mathit{action}(a,d),t)))$

$[A_5]$ $\mathbf{C}(\forall a,f,t,t' \ . \ \mathbf{B}(a,\mathit{holds}(f,t)) \wedge \mathbf{B}(a,t<t') \wedge \neg \mathbf{B}(a,\mathit{clipped}(t,f,t')) \Rightarrow \mathbf{B}(a,\mathit{holds}(f,t')))$

$\mathcal{CA}$: 11 Axioms (Initially)



Plan

P2B

K2B $\forall a[\mathbf{K}_a\phi \rightarrow (\mathbf{B}_a\phi \wedge \mathbf{B}_a\exists\Phi\exists\alpha(\Phi \rightsquigarrow_{\alpha/\pi} \phi)]$

Intro

Incorr $\forall a\forall t\forall F[(F\, is\ contingent\ \wedge F \in C'') \rightarrow (\Box\mathbf{B}(a,t,Fa) \rightarrow Fa)]$

Ess

¬CompE

Irr

Free

CCaus

TheI

$[A_1]$ $\mathbf{C}(\forall f,t\ .\ initially(f) \wedge \neg clipped(0,f,t) \Rightarrow holds(f,t))$

$[A_2]$ $\mathbf{C}(\forall e,f,t_1,t_2\ .\ happens(e,t_1) \wedge initiates(e,f,t_1) \wedge t_1 < t_2 \wedge \neg clipped(t_1,f,t_2) \Rightarrow holds(f,t_2))$

$[A_3]$ $\mathbf{C}(\forall t_1,f,t_2\ .\ clipped(t_1,f,t_2) \Leftrightarrow [\exists e,t\ .\ happens(e,t) \wedge t_1 < t < t_2 \wedge terminates(e,f,t)])$

$[A_4]$ $\mathbf{C}(\forall a,d,t\ .\ happens(action(a,d),t) \Rightarrow \mathbf{K}(a, happens(action(a,d),t)))$

$[A_5]$ $\mathbf{C}(\forall a,f,t,t'\ .\ \mathbf{B}(a, holds(f,t)) \wedge \mathbf{B}(a, t < t') \wedge \neg\mathbf{B}(a, clipped(t,f,t')) \Rightarrow \mathbf{B}(a, holds(f,t')))$

$\mathbf{C}$ `SpecRel`

$\mathbf{C}$ $\mathcal{EC}$

Basic Idea, Intuitively Put

The level of (cognitive) intelligence of an agent (artificial or natural) at a time is a list of tuples (= matrix) giving eg the size of logical depth of multiple measures for each cognitive operator (i.e. for **K**, **B**, **P**, …).

$$\langle [\![\mathbf{K}, 1]\!], [\![\mathbf{K}, 2]\!], \ldots, [\![\mathbf{K}, 5]\!], \ldots \rangle$$

Basic Idea, Intuitively Put

The level of (cognitive) intelligence of an agent (artificial or natural) at a time is a list of tuples (= matrix) giving eg the size of logical depth of multiple measures for each cognitive operator (i.e. for **K**, **B**, **P**, …).

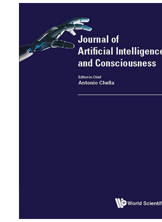
$$\langle [\![\mathbf{K}, 1]\!], [\![\mathbf{K}, 2]\!], \ldots, [\![\mathbf{K}, 5]\!], \ldots \rangle$$

depth of knowledge

size of supporting proof/argument

depth of quantification within outermost knowledge operator

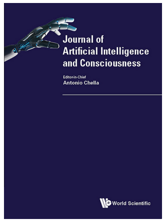
Journal of Artificial Intelligence and Consciousness | Vol. 07, No. 02, pp. 155-181 (2020)



The Theory of Cognitive Consciousness, and Λ (Lambda)

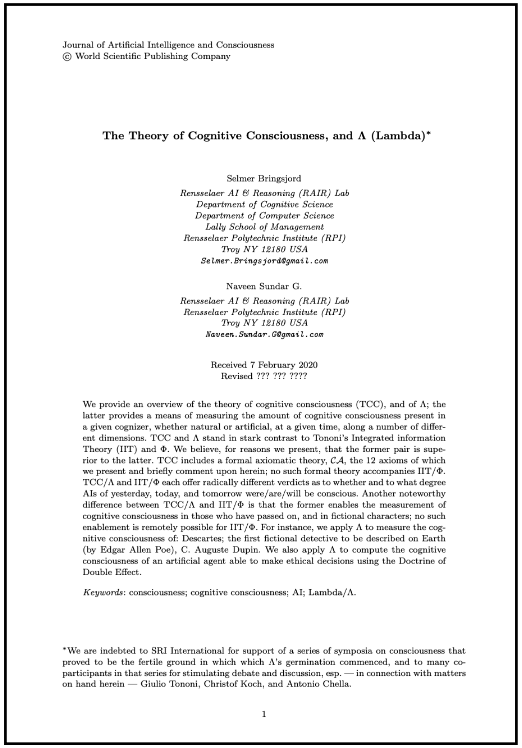
Selmer Bringsjord and G. Naveen Sundar

Journal of Artificial Intelligence and Consciousness | Vol. 07, No. 02, pp. 155-181 (2020)



The Theory of Cognitive Consciousness, and Λ (Lambda)

Selmer Bringsjord and G. Naveen Sundar



Journal of Artificial Intelligence and Consciousness

The Theory of Cognitive Consciousness, and Λ (Lambda)*

Selmer Bringsjord
Rensselaer AI & Reasoning (RAIR) Lab
Department of Cognitive Science
Department of Computer Science
Lally School of Management
Rensselaer Polytechnic Institute (RPI)
Troy NY 12180 USA
Selmer.Bringsjord@gmail.com

Naveen Sundar G.
Rensselaer AI & Reasoning (RAIR) Lab
Rensselaer Polytechnic Institute (RPI)
Troy NY 12180 USA
Naveen.Sundar.G@gmail.com

Received 7 February 2020
Revised ??? ??? ????

We provide an overview of the theory of cognitive consciousness (TCC), and of Λ; the latter provides a means of measuring the amount of cognitive consciousness present in a given cognizer, whether natural or artificial, at a given time, along a number of different dimensions. TCC and Λ stand in stark contrast to Tononi's Integrated information Theory (IIT) and Φ. We believe, for reasons we present, that the former pair is superior to the latter. TCC includes a formal axiomatic theory, $\mathcal{CA}$, the 12 axioms of which we present and briefly comment upon herein; no such formal theory accompanies IIT/Φ. TCC/Λ and IIT/Φ each offer radically different verdicts as to whether and to what degree AIs of yesterday, today, and tomorrow were/are/will be conscious. Another noteworthy difference between TCC/Λ and IIT/Φ is that the former enables the measurement of cognitive consciousness in those who have passed on, and in fictional characters; no such enablement is remotely possible for IIT/Φ. For instance, we apply Λ to measure the cognitive consciousness of: Descartes; the first fictional detective to be described on Earth (by Edgar Allen Poe), C. Auguste Dupin. We also apply Λ to compute the cognitive consciousness of an artificial agent able to make ethical decisions using the Doctrine of Double Effect.

Keywords: consciousness; cognitive consciousness; AI; Lambda/Λ.

*We are indebted to SRI International for support of a series of symposia on consciousness that proved to be the fertile ground in which which Λ's germination commenced, and to many co-participants in that series for stimulating debate and discussion, esp. — in connection with matters on hand herein — Giulio Tononi, Christof Koch, and Antonio Chella.

Journal of Artificial Intelligence and Consciousness | Vol. 07, No. 02, pp. 155-181 (2020)

The Theory of Cognitive Consciousness, and Λ (Lambda)

Extending Measures from $\mathcal{L}^0$ to $\mathcal{L}$

$$\mu_\omega(\phi) = \begin{cases} \mu(\phi) & \text{if } \phi \in \mathcal{L}^0 \\ \max_\psi \mu_\omega(\psi) + 1 & \text{if } \phi \equiv \omega_i\,[a_1, t_1, \ldots \psi \ldots] \end{cases}$$

For example, let μ count the number of predicate symbols in a formula.

Example

$$\mu(Happy(john)) = 1$$
$$\mu_\omega(Happy(john)) = 1$$
$$\mu_\omega\Big(\mathbf{B}(mary, t_2, Happy(john))\Big) = 2$$

For any agent a, we want to look at the new complexity the agent introduces that is above any input complexity. For this, we introduce $\Delta : 2^{\mathcal{L}} \times 2^{\mathcal{L}} \rightarrow 2^{\mathcal{L}}$ operator that computes differences between two sets of formulae. This can be simply the set-difference operator. For convenience, let $\omega_j[\Gamma]$ denote the subset of formulae with operators ω_j in Γ:

$$\omega_j[\Gamma] = \{\phi \mid \ \phi \equiv \omega_j[\ldots] \text{ and } \phi \in \Gamma \text{ or } \phi \text{ a subformula } \in \Gamma\}$$

Given a set of measures $\{\mu^0, \ldots, \mu^N\}$ and a set of modal (or cognitive) operators $\{\omega_0, \ldots, \omega_M\}$, we define Λ as a function mapping an agent at a time point to a matrix $\mathbb{N}^{M \times N}$:

$$\Lambda : A \times T \rightarrow \mathbb{N}^{M \times N}$$

Definition of Λ

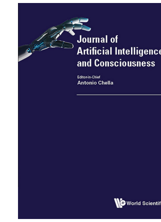
$$\Lambda(a,t)_{i,j} = \max_\phi \Big\{\mu^i(\phi) \mid \phi \in \Delta\Big(\omega_j\big[o(a,t)\big], \omega_j\big[i(a,t)\big]\Big)\Big\}$$

Example 2

Let us consider two modal operators $\{\mathbf{B}, \mathbf{D}\}$ and the following base measures μ^0 which measures quantificational complexity via Σ or Π measures, μ^1 which counts the total number of predicate symbols (not a count of unique predicate symbols), and μ^2 which counts the number of distinct time expressions. This gives $\Lambda : A \times T \rightarrow \mathbb{N}^{2\times 3}$. At some timepoint t, let an agent a have the following $\Delta(o(a,t), i(a,t)) = \{\mathbf{B}(\phi_1), \mathbf{D}(\phi_2)\}$

Journal of Artificial Intelligence and Consciousness | Vol. 07, No. 02, pp. 155-181 (2020)

The Theory of Cognitive Consciousness, and Λ (Lambda)



Extending Measures from $\mathcal{L}^0$ to $\mathcal{L}$

$$\mu_\omega(\phi) = \begin{cases} \mu(\phi) & \text{if } \phi \in \mathcal{L}^0 \\ \max_\psi \mu_\omega(\psi) + 1 & \text{if } \phi \equiv \omega_i\,[a_1, t_1, \ldots \psi \ldots] \end{cases}$$

For example, let μ count the number of predicate symbols in a formula.

Example

$$\mu\left(Happy\left(john\right)\right) = 1$$
$$\mu_\omega\left(Happy\left(john\right)\right) = 1$$
$$\mu_\omega\left(\mathbf{B}\left(mary, t_2, Happy(john)\right)\right) = 2$$

For any agent a, we want to look at the new complexity the agent introduces that is above any input complexity. For this, we introduce $\Delta : 2^{\mathcal{L}} \times 2^{\mathcal{L}} \rightarrow 2^{\mathcal{L}}$ operator that computes differences between two sets of formulae. This can be simply the set-difference operator. For convenience, let $\omega_j[\Gamma]$ denote the subset of formulae with operators ω_j in Γ:

$$\omega_j[\Gamma] = \{\phi \mid \ \phi \equiv \omega_j[\ldots] \text{ and } \phi \in \Gamma \text{ or } \phi \text{ a subformula } \in \Gamma\}$$

Given a set of measures $\{\mu^0, \ldots, \mu^N\}$ and a set of modal (or cognitive) operators $\{\omega_0, \ldots, \omega_M\}$, we define Λ as a function mapping an agent at a time point to a matrix $\mathbb{N}^{M \times N}$:

$$\Lambda : A \times T \rightarrow \mathbb{N}^{M \times N}$$

Definition of Λ

$$\Lambda(a,t)_{i,j} = \max_\phi \left\{\mu^i(\phi) \mid \phi \in \Delta\left(\omega_j\left[o(a,t)\right], \omega_j\left[i(a,t)\right]\right)\right\}$$

Example 2

Let us consider two modal operators $\{\mathbf{B}, \mathbf{D}\}$ and the following base measures μ^0 which measures quantificational complexity via Σ or Π measures, μ^1 which counts the total number of predicate symbols (not a count of unique predicate symbols), and μ^2 which counts the number of distinct time expressions. This gives $\Lambda : A \times T \rightarrow \mathbb{N}^{2\times 3}$. At some timepoint t, let an agent a have the following $\Delta(o(a,t), i(a,t)) = \{\mathbf{B}(\phi_1), \mathbf{D}(\phi_2)\}$

$$\phi_1 \equiv \neg\forall a : Happy(a,t); \qquad \phi_2 \equiv \forall b : \neg Hungry(b,t) \rightarrow Happy(b,t)$$

Applying the measures:

$$\mu^o(\phi_1) = 1, \mu^1(\phi_1) = 1; \mu^2(\phi_1) = 1$$
$$\mu^o(\phi_2) = 1; \mu^1(\phi_2) = 2; \mu^2(\phi_2) = 1$$

Giving us:

$$\Lambda(a,t) = \begin{pmatrix} 1\,1\,1 \\ 1\,2\,1 \end{pmatrix}$$

6.1. *Some Distinctive Properties of* Λ *(vs.* Φ*)*

Here are some properties of the Λ framework of potential interest to our readers:

Non-Binary Whereas Φ is such that an agent either is or is not (P-) conscious, cognitive consciousness as measured by Λ admits of a fine-grained range of the *degree* of cognitive consciousness.

Zero Λ for Some Animals and Machines Animals such as insects, and computing machines that are end-to-end statistical/connectionist "ML," have zero Λ, and hence cannot be cognitively conscious. In contrast, as emphasized to Bringsjord in personal conversation,[6] Φ says that even lower animals are conscious.

Human-Nonhuman Discontinuity Explained by Λ From the computational/AI point of view, cognitive scientists have taken note of a severe discontinuity between *H. sapiens sapiens* and other biological creatures on Earth [Penn *et al.*, 2008], and the sudden and large jump in level of Λ from (say) chimpanzees and dolphins to humans is in line with this observation. It's for instance doubtful that any nonhuman animals are capable of reaching third-order belief; hence $\Lambda[\mathbf{B}, 0] = n$, where $n \geq 3$, for any nonhuman animal, is impossible. In stark contrast, each of us believes that you, the reader, believe that we believe that San Francisco is located in California.

Human-Human Discontinuity Explained by Λ A given neurobiologically normal human, over the course of his or her lifetime, has very different cognitive capacity. E.g., it's well-known that such a human, before the age of four or five, is highly unlikely to be able to solve what has become known as the *false-belief task* (or sometimes the *sally-anne task*), which we denote by 'FBT.' From the point of view of Λ, the explanation is simply that an agent with insufficiently high cognitive consciousness is incapable of solving such a task; specifically, solving FBT requires an agent to have

[6]With Tononi and C. Koch, SRI T&C Series.

Btw, what about eg (End-to-End) “Deep Learning”?

Btw, what about eg (End-to-End) “Deep Learning”?

$$\frac{|\mathbf{K}_{\mathfrak{h}}\Phi|}{|\alpha|} \gg \frac{|\mathbf{K}_{\mathfrak{dla}}\Phi|}{|D+C|}$$

Btw, what about eg (End-to-End) "Deep Learning"?

$$\frac{|\mathbf{K}_{\mathfrak{h}}\Phi|}{|\alpha|} \gg \frac{|\mathbf{K}_{\mathfrak{dla}}\Phi|}{|D + C|}$$

$$\Lambda[\text{GPT-3}] = 0$$

What is the level of consciousness (= Λ value) enjoyed by this self-conscious robot?

https://motherboard.vice.com/en_us/article/mgbyvb/watch-these-cute-robots-struggle-to-become-self-aware

Theorem-Sketch: UCI is discontinuous (human vs. nonhuman animal; and human vs. AI).

“**Theorem**”: C-con., as measured by Λ, unlike P-con. as measured by Φ, is *dis*continuous.

Theorem-Sketch: UCI is discontinuous (human vs. nonhuman animal; and human vs. AI).

Med nok penger, kan logikk løse alle våre problemer.