Truth and Justification in Knowledge Representation

Andrei Rodin[1,2], Serge Kovalyov[3]

[1] Institute of Philosophy RAS, 12/1 Goncharnaya Str., Moscow, 109240, Russian Federation

[2] HSE, 20 Myasnitskaya Ulitsa, Moscow, 101000, Russian Federation
avrodin@hse.ru

[3] ICS RAS, 65 Profsoyuznaya street, Moscow 117997, Russian Federation
kovalyov@sibnet.ru

ABSTRACT: *While the traditional philosophical epistemology stresses the importance of distinguishing knowledge from true beliefs, the formalisation of this distinction with standard logical means turns out to be problematic. In Knowledge Representation (KR) as a Computer Science discipline this crucial distinction is largely neglected. A practical consequence of this neglect is that the existing KR systems store and communicate knowledge that cannot be verified and justified by users of these systems without external means. Information obtained from such systems does not qualify as knowledge in the sense of philosophical epistemology.*

Recent advances in the research area at the crossroad of the computational mathematical logic, formal epistemology and computer science open new perspectives for an effective computational realisation of justificatory procedures in KR. After exposing the problem of justification in logic, epistemology and KR, we sketch a novel framework for representing knowledge along with relevant justificatory procedures, which is based on the Homotopy Type theory (HoTT). This formal framework supports representation of both propositional knowledge, aka knowledge that, and non-propositional knowledge, aka knowledge-how or procedural knowledge. The default proof-theoretic semantics of HoTT allows for combining the two sorts of represented knowledge at the formal level by interpreting all permissible constructions as justification terms (witnesses) of associated propositions.

1. Concept of Knowledge

1.1 Knowledge according to the Philosophical Epistemology
JTB and Gettier Problem The current philosophical discussion on the concept of knowledge focuses on the so-called JTB

theory of knowledge according to which knowledge is Justified True Belief. The JTB theory dates back to Plato and, in modern terms, states that subject S knows that p (where p is a proposition) just in case the following three conditions are satisfied [1]:

1. p is true

2. S believes that p

3. S is justified in believing that p.

Leaving the psychological concept of belief aside of our present analysis we would like to stress the following features of JTB theory:

• JTB identifies knowledge with knowledge of certain proposition or propositions; this type of knowledge is conventionally referred to as propositional knowledge aka knowledge-that.

• JTB assumes that the truth-value of a given proposition is determined wholly independently of one's knowledge of this proposition. Such an account of truth has a long tradition in logic and has been strongly defended, among other people, by Gottlob Frege. We shall shortly show that this conception of truth is not commonly accepted in the philosophical logic.

• According to JTB, a true belief, i.e., one's belief in certain true proposition, by itself does not constitute knowledge. A missing element is justification. Assuming that a mathematical proof is a special form of justification, for a motivating example think of Bob who is able to state the Pythagorean theorem (provided she understands its meaning and believes it is true) and Alice who is also able to prove it. In terms of JTB theory Alice knows the theorem but Bob doesn't. What is at stake here is not the linguistic meaning of \know" but the difference between the two sorts of epistemic states, viz. knowledge and (true) belief (or however one may prefer to call them).

A major part of the mainstream discussion on and about JTB concerns the socalled Gettier Problems. In 1963 Edmund Gettier published a highly influential paper [2] where he showed using some linguistic examples that the concept of justification involved into JTB is very problematic: a subject can be compelled to belief that p, where p is true proposition, by certain reason r, which she relates to p by a mere mistake; allegedly such "false reasons" cannot be ruled out by the JTB theory, so this theory is at best incomplete and at worst wholly wrong. Without being able to discuss here Gettier-style epistemological problems systematically we would like to express our general take on this issue: in our view, the core problem here is that the concept of justification unlike that of truth is not adequately accounted for by standard logical tools. On the contrary to a popular opinion it also concerns mathematical proofs. However during last decades there was a significant progress towards this goal some elements of which are described in what follows.

Since JTB accounts only for the propositional knowledge certain authors argue that it doesn't cover the concept of knowledge in its full extent leaving aside an irreducibly procedural knowledge aka knowledge-how. We endorse this view and remark after Ryle [3] that knowing how to make logical inferences and otherwise justify one's beliefs is (irreducibly) procedural rather than propositional in its character. Another challenge for JTB comes from the constructive tradition in logic, which tightly relates truth and knowledge by identifying truth of proposition with the existence of its proof (evidence). From the constructive point of view the tripartition of knowledge into (i) true proposition, (ii) one's belief in this proposition and (iii) justification of this belief is hardly tenable because here truth of a given proposition requires its justification (proof) in some form at the first place. Notice that even if this constructive approach is incompatible with JTB in its usual form, it shares with JTB the notion according to which justification is a necessary element of knowledge. In fact the constructive approach takes justification to be even more important by making it constitutive for truth and logic itself. M. Cohen and E. Nagel express this view on logic when they describe its purpose, in full generality, as the "determination of the best available evidence" [4].

---

[4] Here is the full quote:

[T]he constant and universal feature of science is its general method, which consists in the persisting search for truth, constantly asking: Is it so? To what extent is it so? Why is it so? [...] And this can be seen on reflection to be the demand for the best available evidence, the determination of which we call logic. Scientific method is thus the persistent application of logic as the common feature of all reasoned knowledge [4, p. 192]

1.2 Knowledge in Computer Science and Information Technology

Knowledge Representation (KR) sometimes also referred to as Knowledge Representation and Reasoning is an established discipline in Computer Science and a developing information technology. Obviously one should not expect to find in KR literature a thorough analysis of the knowledge concept, which can be more appropriate in the philosophical literature. Nevertheless authors of some monographs and textbooks in KR provide informal descriptions of basic concepts of the discipline including that of knowledge and reasoning [5], [6], [7]. Remarkably, none of such descriptions found by us in the CS literature mentions the standard epistemological condition according to which knowledge needs to be justified.

This observation squares with another one. In 1980-ies the philosophical term "ontology" began its independent life in CS and since 1995 the latest [8] has been used systematically as a standard technical term and concept in the KR design and research. In philosophy the term "ontology" refers to the problematic area of research and reflection that concerns, to use Aristotle's famous word, the being qua being or, in more modern terms, general features of all entities in their mere capacity of being existent: it includes classifications of entities into different sorts (e.g., objects, events and their properties) and the like. Ontologies used in KR are computationally implemented formal semantic frameworks for representing objects and their mutual relations; knowledge represented in a KR system refers to these objects and relations as its subject-matter, i.e., to what this knowledge is "about".

Despite the fact that KR ontologies lose at some extent their philosophical origins, the difference between the philosophical and the CS concepts of ontology is not dramatic. Formal ontology, which is a philosophical ontology developed with a support of formal logical methods, can be seen as a middle ground that links the traditional philosophical ontology, on the one hand, and the technical concept of ontology used in KR, on the other hand.

What is puzzling here in eyes of a philosopher is the following. A philosophical discipline that covers problems concerning knowledge is called epistemology but not ontology. Just like ontology epistemology is developed, in part, with a support of formal methods; this approach is known as formal epistemology. Yet, the CS discipline that essentially involves the concept of knowledge, viz. KR, for some reason makes use of ontology but not of epistemology.

We don't assume here that CS or any other engineering discipline must respect traditional philosophical distinctions when it borrows philosophical terms and concepts and then modifies them for its own use. However we claim that the above observations point to a real problem, which has a practical dimension. The issue of reliability of information available via electronic communications is widely discussed in special and general literature and since recently is also recognised as an important social and even political problem [9]. The existing data verification technologies are designed for serving developers and administrators of KR systems rather than its regular users, and for this reason don't fully address this problem. In order to make a piece of information obtained by a user of KR system reliable in eyes of this very user (as this is required by the JTB conception of knowledge and by any other conception that takes the issue of justification seriously) a supporting evidence needs to be available to the user herself. We assume here that this evidence also needs to be specific and not reduce to a general assurance that the given KR system is reliable.

A part of the problem, as we see it, is that the standard logical tools such as the first-order Classical logic along with its usual philosophical underpinning leave the epistemic concept of justification aside. The philosophical conceptions of truth and logical reasoning that underline this notion of logic prioritise ontological aspects with respect to epistemological ones. Correspondingly, theoretical prototypes of KR systems, which use this standard logical and semantic framework, essentially use ontologies but don't support the epistemic procedure of justification. If Sundolm is right that epistemic considerations have been systematically neglected in the mainstream logical research until very recently [10], it is a little surprise that they have been also neglected in CS. In the next Section we elaborate on this point providing more details and then propose a remedy.

2. Model-theoretic and Proof-theoretic Semantics

The standard notion of axiomatic theory that stems from Hilbert assumes that:

• A theory is set T of formulas that are interpreted as true statements; such interpretations of formulas are called models of the given theory;

• A theory has a subset of formulas $A \subset T$ called axioms of the given theory;

• A theory comprises set R of syntactic rules, which, in particular, regulate derivations of new T-formulas from some given T-formulas. T - derivations preserve truth in the sense that given any model m of T, they derive from true sentences only true sentences (soundness). T comprises all formulas T - derivable from its axioms (deductive closure).

• T- formulas, which are T- derivable from the axioms (other than the axioms) are called theorems of T. A derivation of theorem from axioms (and by extension also from some intermediate theorems) is called a proof of a given theorem. The standard notation for the derivability of theorem B from axioms $A_1,..., A_n$ in theory T is as follows: $A_1,..., A_n \vdash_T B$.

This familiar scheme involves at least one epistemic term, namely, "proof". However as convincingly, in our view, argues Prawitz the bold identification of proofs with syntactic derivations is not justified [11]. In order to explain the argument we need the following formal notion of logical consequence relation due to Tarski [12]:

Definition 1 T -formula B is called a logical consequence of T- formulas $A_1,..., A_n$, in symbols $A_1,..., A_n \vDash_T B$ just is case every interpretation m that interprets $A_1,..., A_n$ as true sentences also interprets B as true sentence.

Since T is sound (with respect to some fixed class of its interpretations), every T- theorem B derived from T-axioms $A_1,..., A_n$ is a logical consequence of these axioms. Prima facie this observation justifies the idea that a formal derivation of B represents its logical inference from T- axioms, which qualifies as its T- proof. Prawitz argues to the contrary. Even if it is the case that a given syntactic derivation faithfully represents a truth-preserving logical inference, nothing guarantees in this setting that the same symbolic representation allows one to see that truth is preserved. A further problem concerns the specific Hilbertian notion of axiom and the related Tarskian notion of truth-in-a-model, which has little to do with the traditional notion of axiom as a self-evident truth. The concept of evidence is wholly alien to this approach and ruled out as psychological and hence logically irrelevant. This makes a sharp contrast with the aforementioned Cohen & Nagel's conception of logic where the notion of evidence has a central role [4]. What is an epistemic value, if any, of a formal proof in the above technical sense remains unclear.

At the same time the above logico-semantic framework can be straightforwardly related to ontology via the following principle known as the truthmaker realism (TMR):

Given a true statement there exists an entity (or entities) that make (s) this statement true. [13]

Once one accepts TMR the notion of formal ontology readily suggests itself as a useful formal semantic tool, which helps one to supplement, and in many applications even to replace, the talk of models and interpretations by talks about some familiar entities that a given theory is supposed to account for. The Tarski-style formal semantics helps to make this ontological talk formal and rigorous. This is a pragmatic reason to accept some form of TMR and the notion of formal ontology, which may convince even those people, including KR developers, who are not interested in traditional philosophical debates about being and existence. One does not need to explore deep philosophical questions about being in order to use formal ontologies as semantic tools. This explains why the notion of formal ontology became useful and popular in the AI research.

However, as we have already stressed, the neglect of epistemic considerations in the foundations of the above logico-semantic framework and, more specifically, the lack of satisfactory formal treatment of justification, rises a problem, which is not only theoretical but also practical. Once the theoretical and practical significance of justification is recognised, it becomes clear that the standard logical and semantical tools are not sufficient for developing theoretical prototypes of reliable KR systems.

In the philosophical and mathematical logic this epistemological problem is well recognised and understood by a part of the professional community. There is presently a number of tentative solutions on the market. A systematic formal treatment of Justification Logic with explicit justificatory terms is given in the new monograph by S. Artemov and M. Fitting [14]. A variety of approaches that attempt to supplement or replace the standard model-theoretic logical semantics (MTS) outlined above by some version of alternative epistemically relevant semantic is now grouped under the header of proof-theoretic semantics (PTS) [15]. It is remarkable that many versions of PTS are more \computer-friendly", i.e., more apt for computer implementation, than their MTS analogues because they give semantic values directly to syntactic rules and procedures rather than only to formulas. A systematic overview of this actively developing area of research is out of the scope of this paper. In the next Section we only briey describe a formal theory that belongs to the PTS family (albeit arguably goes beyond PTS in some essential aspects) and propose it to the role of novel formal semantic framework for KR.

3. Homotopy Type Theory as KR Framework

3.1 MLTT, HoTT and Their Proof-Theoretic Semantics

Homotopy Type theory (HoTT) is a growing family of type theories with dependent types, which are interpreted (more or less formally) in terms of Homotopy theory, which is a part of Algebraic Topology[5]. Such homotopical interpretations of type theories were discovered independently by Steven Awodey and Vladimir Voevodsky in mid 2000-ies. We consider here the standard HoTT presented in [17], which uses the syntax of Martin-Lnof's Type theory (MLTT) [16] extended with the single Univalence Axiom, which is out of the scope of the present discussion. This version of HoTT preserves the core proof-theoretic semantics of the original MLTT and extends it with a new homotopy semantics. We analyse the relationships between the original MLTT semantics and the HoTT semantics and attempt to make their combination coherent.

MLTT is a rule-based formal system that comprises no axiom. Its basic formulas are called judgements and interpreted accordingly. MLTT comprises four basic forms of judgements.

(i) $A : TYPE$;

(ii) $A \equiv_{TYPE} B$;

(iii) $a : A$;

(iv) $a \equiv_A a'$

In words (i) says that A is a type, (ii) that types A and B are the same, (iii) that a is a term of type A and (iv) that a and a' are the same term of type A. We now leave (i) and (ii) aside and provide more details on (iii) and (iv).

Martin-Lof offers four different informal readings of (iii) [16, p. 5]:

1. a is an element of set A

2. a is a proof (construction) of proposition A ("propositions-as-types")

3. a is a method of fulfilling (realizing) the intention (expectation) A

4. a is a method of solving the problem (doing the task) A (BHK semantics)

The author argues that these interpretations of judgement form (iii) not only share a logical form but also are closely conceptually related despite of their different linguistic appearance.

Let us now turn to judgement form (iv). It says that terms a, a', both of the same type A, are equal. This equality is called judgemental or definitional and does not qualify as a proposition; the corresponding propositional equality writes as $a =_A a'$ and counts as a type on its own ($a =_A a' : TYPE$) called an identity type. In accordance to reading (2) of judgement form (iii) a term of identity type is understood as a proof (also called a witness or evidence) of the corresponding proposition. MLTT validates the rule according to which a judgemental equality entails the corresponding propositional equality:

$$\frac{a \equiv_A a'}{refl_a : \equiv_A a'}$$

where $refl_a$ is the canonical proof of proposition $a \equiv_A a'$.

The extensional version of MLTT also validates the converse rule called the equality reflection rule:

$$\frac{p : a \equiv_A a'}{a \equiv_A a'}$$

HoTT draws on an intensional version of MLTT that does not use such a principle and allows for multiple proofs of the same propositional equality.

Let now *p, q* be two judgmentally different proofs of proposition saying that two terms of a given type are equal:

$$p, q : P =_T Q$$

It may be the case that *p, q*, in their turn, are propositionally equal, and that there are two judgmentally different proofs *p′, q′* of this fact:

$$p', q' : p = P=_T Q \, q$$

This and similar multi-layer syntactic constructions in MLTT can be continued unlimitedly. Before the rise of HoTT it was not clear that this syntactic feature of the intensional MLTT can be significant from a semantic point of view. However it became the key point of the homotopical interpretation of this syntax. Under this interpretation.

• Types and their terms are interpreted, correspondingly, as spaces and their points;

• Identity proofs of form *p, q : P =_T Q* are interpreted as paths between points *P, Q* of space *T*;

• Identity proofs of the second level of form *p′, q′ : p = P=_T Q q* are interpreted as homotopies between paths *p, q*;

• All higher identity proofs are interpreted as higher homotopies;

Recall that path p between points *P, Q* of topological space *T* is continuous map $p : [0, 1] \to T$ such that $p(0) = P$ and $p(1) = Q$. Intuitively a path can be thought of as a trajectory of moving test point where the real interval $[0,1]$ represents time. In a more abstract presentation the real unit interval $[0,1]$ is replaced by abstract unit object *I*. Homotopy *h* between paths *p, q* is continuous map $h : [0, 1]^2 \to T$ such that $h(t, 0) = p(t)$ and $h(t, 1) = q(t)$; intuitively it can be thought of as a "path between paths" or a continuous transformation of path *p* into path *q*. Higher homotopies are defined similarly. For a modern introduction into the basic Homotopy theory see [19].

The homotopical interpretation makes the complex structure of identity types in the intensional MLTT surveyable and suggests a revision of the original semantics of MLTT by distinguishing between propositional and non-propositional types on the syntactic level. According to this new point of view not every type can be interpreted either as a proposition or as a set but each of these two interpretations is admissible only for types of appropriate sorts. More precisely, consider the following

Definition 2 Space aka homotopy type *S* is called contractible or space (type) of h-level (-2) when there is point *p : S* connected by a path with each point *x : A* in such a way that all these paths are homotopic (i.e., there exists a homotopy between any two such paths).

Definition 3 We say that *S* is a space of h- level *n + 1* if for all its points *x, y* path spaces *x = S y* are of h- level *n*.

These definitions gives rise to the following stratification of types/spaces in HoTT by their *h*- levels:

• *h*- level (-2): single point *pt*;

• *h*- level (-1): the empty space $\varnothing$; and the point *pt*: truth-values aka (mere) propositions;

• *h*- level 0: sets aka discrete point spaces: comprise no non-contractible paths;

• *h*- level 1: at path groupoids : comprise paths but no non-contractible surfaces;

• *h*- level 2: 2-groupoids : comprise paths and surfaces but no non-contractible volumes;

• ....

• *h*- level ω : ω- groupoids.

Space S_n of h- level n can be transformed into a space $[S]_k$ of h- level $k < l$ via its k- truncation, which can be informally described as a forced identification of all homotopies (paths) of all levels higher than k. In particular, the (-1)- truncation $[S]_{-1}$ of any given space S brings point pt when S is not empty and brings the empty space $\varnothing$; otherwise.

The notion of truncation allows for interpreting type $[S]_{-1}$ as a proposition and the original type S as a (h- stratied) space of proofs of this proposition: $[S]_{-1}$ is true when it has a proof in S and is false otherwise. Assuming that the scope of logic restricts to propositional types one can now describe higher types and their terms as being extra-logical. However the homotopic semantics of the extra-logical terms still qualifies as proof-theoretic because such terms serve as proof terms of certain propositions.

3.2 How to use HoTT for KR Purposes
If different terms of the same type are not distinguished then HoTT is functioning as a constructive propositional logic with explicit proof terms, which in this case can be also called internalised truth-values. If the (in)equalities of terms are taken into account only up to the set level (which means that distinctions between different paths between the same terms, i.e., between \dierent ways of being equal", are ignored) then HoTT functions as a constructive first-order calculus with internalised (constructive) sets that already provides more information about its proof terms. These sets are constructively internalised in the sense that they are represented here with syntactic constructions available in HoTT itself rather than introduced with a help of some external meta-theoretical tools as this is done in case of standard Tarski semantics for the Classical first-order logic.

This feature alone demonstrates a potential of HoTT as a representational framework: it supports representation of propositions along with objects that those propositions are "about"; the same terms can be also described as truthmakers of their base propositions, their evidences or their proofs. Accordingly, HoTT represents a propositional knowledge (since the true represented propositions are evidenced) along with an associated procedural knowledge, viz., the knowledge of how to construct for the given proposition its evidences aka proofs. Such a justificatory procedure for propositional knowledge has its formal dual in the form of verification of the corresponding procedural knowledge. In this case the epistemic goal is not to justify a propositional belief but to assure that an accomplished construction has some required properties. Think of technological processes which certain desired outcomes, which needs to be checked and verified. Since this difference in epistemic goals does not affect the basic semantics of HoTT, our proposed approach applies to both these sorts of tasks.

Higher levels of the homotopy ladder provide more expressive power for representing objects and spaces where these objects live. The (at) groupoid spaces (h- level 1) already allow for representing certain non-trivial topological features of the base spaces. Leaving for another occasion a study of possible applications of topological concepts in KR we would like to stress here its intuitive appeal. This is not a minor issue when we are talking about possible ways to justify knowledge obtained via a KR system, which is supposed to be available to a regular user. A HoTT-based approach has been already successfully used for an automated verification of non-trivial mathematical proofs [21]. An advantage of this approach over other approaches in the automated proof verification is that the homotopical interpretation allows a mathematician to express her reasoning with a commented program code or a pseudo-code without giving up the usual intuitive support of this reasoning. This specic feature of HoTT might be helpful for designing a format for human-readable evidences or certificates that a hypothetical KR system could produce in order to justify the supplied knowledge in eyes of its user. A toy example of HoTT-based representation used outside the pure mathematics is found in [20].

We summarise our explanation of relevance and possible advantages of using HoTT as a formal KR framework that supports justification as follows:

1. HoTT admits the constructive epistemically-laden proof-theoretic semantics intended by Martin-Lnöf's Type for MLTT (in a slightly modified form).

2. The cumulative h- hierarchy of types made explicit via the homotopical interpretation supports the distinction between propositional, set-level and higher-level types. This distinctive feature of HoTT supports formal constructive representation of objects (of various levels) and propositions "about" these objects within the same framework. Each such object serves as a witness/truthmaker for proposition obtained via the propositional truncation of type where the given object belongs.

3. HoTT comprises a system of formal rules, which are interpreted as logical rules at the propositional h- level and as rules for

object-construction at all higher levels. This feature of HoTT supports representation various extralogical procedures (such as material technological procedures) keeping track of the corresponding logical procedures at the propositional level of representation.

4. HoTT/MLTT is computer-friendly, i.e., computationally implementable. Fragments of HoTT/MLTT have been implemented in proof-assistant Coq, program languages AGDA, LEAN and some other products.

5. HoTT-constructions admit intuitive spatial (homotopical) interpretations that may be used for facilitating human-computer interactions.

4. Conclusion

During the last decade KR technologies have been enriched with approaches based on the Big Data analysis, Machine Learning and artificial Neural Networks. According to a radical opinion, these new approaches make more traditional logical approaches based on the explicit representation of facts and rules hopelessly outdated and irrelevant. We disagree. Because of their possible unpredictable behaviour [22] Neural Networks and other tools of Big Data analysis can significantly enrich but not replace logical approaches and logical tools in KR.

At the same time we agree that standard logical architectures and formal ontologies, which are presently used in KR, don't provide a sufficient theoretical background for KR because they have no epistemological content. In this paper we explained the relevance of epistemological considerations in logic and KR and then pointed to some recent advances in mathematical logic, more specifically discussing the Homotopy Type theory, that may allow to use logical approaches in KR more effectively.

References

[1] Ichikawa, J. J., Steup, M. (2018). The Analysis of Knowledge, The Stanford Encyclopedia of Philosophy (Summer 2018 Edition), Edward N. Zalta (ed.), URL = https://plato.stanford.edu/archives/sum2018/entries/knowledge-analysis/

[2] Gettier, E. L. (19363). Is Justified True Belief Knowledge?, Analysis, 23(6) 121123 (1963), https://doi.org/10.2307/3326922

[3] Ryle, G. (1945). Knowing How and Knowing That: The Presidential Address, *In*: Proceedings of the Aristotelian Society, New Series, 46, 1-16 (1945-46).

[4] Nagel, E., Cohen, M. R. (1934). An Introduction to Logic and Scientific Method, Routledge.

[5] Jakus, G. (2013). Concepts, Ontologies and Knowledge Representation, Springer.

[6] Abraham, A., Grosan, C. (2011). Intelligent Systems : A Modern Approach, Springer.

[7] Lakemeyer, G., Nebel, B. (1994). Foundations of Knowledge Representation and Reasoning, Springer.

[8] Gruber, Th. R. (1995). Toward principles for the design of ontologies used for knowledge sharing?, *International Journal of Human-Computer Studies*, 43(5-6) 907-928.

[9] Fuller, S. (2018). Post-Truth: Knowledge as a Power Game, Cambridge University Press.

[10] Sundholm, G. (2018). The Neglect of Epistemic Considerations in Logic: the Case of Epistemic Assumptions, Topoi 38 (3) 551-559 (2018) https://doi.org/10.1007/s11245- 017-9534-0 (Open Access)

[11] Prawitz, D. (1974). On the Idea of the General Proof Theory, Synthese, 27(1-2) 63-77.

[12] Tarski, A. (1983). On the Concept of Logical Consequence, *In*: Logic, Semantics, Metamathematics, Hackett Publ., 1983, p. 409-420

[13] Smith, B. (1999). Truthmaker realism, *Australasian Journal of Philosophy*, 77 (3) 274- 291.

[14] Artemov, S., Fitting. (2019). Justification Logic: Reasoning with Reasons, Cambridge University Press.

[15] Piecha, Th., Schroeder-Heister, P. (2015). Advances in Proof-Theoretic Semantics, Springer.

[16] Martin-Lnof, P. (1984). Intuitionistic Type Theory, BIBLIOPOLIS.

[17] Univalent Foundations Program: Homotopy Type Theory, IAS Princeton, 2013 URL = https://homotopytypetheory.org/

book/ (Open Access)

[18] Rodin, A. (2019). Models of HoTT and the Constructive View of Theories, *In*: Centrone, S., Kant, D. and Sarikaya, D. (Eds.) Reections on the Foundations of Mathematics: Univalent Foundations, Set Theory and General Thoughts, Springer 2019, p. 189-220

[19] Strom, J. (2011). Modern Classical Homotopy Theory, American Mathematical Society.

[20] Rodin, A. (2017). Venus Homotopically, *IfCoLog Journal of Logics and their Applications*, 4(4) 1427-1446 (2017, open access)

[21] Grayson, D. R. (2017). An introduction to univalent foundations for mathematicians, *Bulletin of the American Mathematical Society* 55 (2017), https://doi.org/10.1090/bull/1616.

[22] Szegedy, C. (2013). *Intriguing Properties of Neural Networks*, CoRR abs/1312.6199.