Constructive Knowledge
(logical and epistemological aspects)

Proceedings of Workshops organized on April 25 and October 5, 2016
in the IP RAS, Moscow

with a support of the Russian Foundation for Humanities grant
16-03-00364

Collected and edited by Andrei Rodin

Institute of Philosophy of Russian Academy of Sciences 2016

Contents

1 Scope and Aim

We call knowledge constructive if it includes an explicit specification of such associated epistemic procedures as capturing, verification, presentation, transmission, revision, and application of the given knowledge. Such a concept of constructive knowledge does not assume the social or cognitive constructivism according to which all knowledge is a social (resp. cognitive) construal independent of any real relation to its object. We study formal logical and epistemological aspects of various epistemic procedures related to knowledge belonging to a wide spectrum of fundamental and applied disciplines. A special focus is made on recent and prospective technologies of knowledge representation and knowledge management.

2 Serge P. Kovalyov (Institute of Control Problems RAS) Machine Intelligence in Engineering of Axiomatic Systems

Rigorous solutions to applied problems are relevant in many areas of technology. Notorious examples include verification of embedded software, axiomatic design of complex manufacturing products, virtual simulation of electronic equipment functioning, etc. [1]. However, in most cases, employing such approaches in real-world projects involves costs that significantly excess the visible useful results. One of the major reasons for this is the poor predictability of activities related to the application of the axiomatic method. It is unclear how to build practically useful axiomatic descriptions of domains without exceeding specified deadlines and budgets. Employing powerful computer tools is considered as a natural approach to solve this problem. Data mining tools are developed to accumulate large amounts of domain data in machine-readable form and process them on computers in order to extract the patterns of interrelations of a general nature that permit extrapolation beyond known data. With the help of domain experts, axioms and inference rules can be selected among patterns to form a deductive domain model as an axiomatic system. To apply axiomatic systems in practical problem solving, other computer tools are developed, viz. provers that are able to verify validity of any statements about the domain by automatic inference. As a result, the promising highly automated mode of application of the axiomatic method emerges, which deserves to be identified as a special branch of knowledge engineering, called engineering of axiomatic systems.

A variety of methods and tools of machine intelligence are used in engineering of axiomatic systems. Known illustrative examples include:

- recovering axioms in the course of automated logical inference on the so-called J-calculus [2];
- Inductive Logic Programming [3];
- domain ontology engineering [4];
- identifying the logical basis of the domain deduction by means of machine learning [5];
- distilling features into concepts (Meta-Interpretive Learning) [6].

The last two of these examples are based on the technology of deep neural networks that currently undergoes very intense development. There is even an opinion that the classical means of knowledge representation, based on explicit symbolic expression of the facts and laws, are hopelessly outdated and will soon be replaced by neural networks that manage implicit knowledge in a distributed form. However, distributed knowledge is unreliable, hard to verify, and prone to misrepresentation. Experiments are known when perfectly recognized images were practically imperceptibly perturbed in a special way, which was calculated in accordance with the "white spots" of the training sample, causing arbitrarily changes in the classifier's output [7]. So neural networks can greatly enhance, but not replace the axiomatic knowledge management tools.

All of these approaches, both formal and neural network based, are intended for use within a single domain. However, the current level of technological development is characterized by a multidisciplinary nature of the manufactured products: the products are complex systems that consist of components taken from several different domains. In order to build a coherent formal description of such a product, mechanisms are needed to integrate domain-specific axiomatic systems into sound holistic bodies of knowledge. Some time ago the ontology engineering "naively" recommended to rely on the public nature of knowledge that allegedly admit direct unification as soon as powerful referential capabilities are provided. However, such recommendation fails in practice due to the presence of antagonistic (ontological in the philosophical sense) contradictions between the diverse participants of the product life cycle.

Fortunately, holistic heterogeneous product descriptions useful for practitioners in systems engineering are typically restricted to certain specific viewpoint, or aspect, that can be identified in each component. The body of axiomatic knowledge about the aspect is sound; hence inter-component contradictions are easily detected, and resolved or left beyond the description. For example, the international standard IEC 81346-1:2009 “Industrial Systems, Installations and Equipment and Industrial Products - Structuring Principles and Reference Designations” specifies such aspects as function, material embodiment, location.

The processes of synthesizing the holistic product descriptions in an aspect are convenient to formally describe using the category-theoretic representation of axiomatic systems which was developed by H. Graves and others [8]. Let C be a category that represent the axiomatic system of the aspect, let I be the scheme (the shape) of the diagram that represents the structure of the complex product, and let $D_i, i \in I$ be the family of categories that represent axiomatic systems of product components. For each component i, the functor $F_i : D_i \to C$ is given that determines the rule to extract the target aspect from descriptions of the component. A particular holistic description of the product in the aspect C is obtained by choice of a family of objects $A_i \in D_i, i \in I$, and a diagram $\Delta : I \to C$ that satisfies the condition $\Delta(i) = F_i(A_i)$ for each $i \in I$.

Procedures of systems engineering are formally described by transformations of such descriptions. To specify and explore such transformations, we construct the category with descriptions as objects. For morphisms of such a category, we naturally employ natural transformations of descriptions’ diagrams induced by morphisms of the components. Specifically, a morphism of a description $((A_i, i \in I), \Delta)$ to a description $((A'_i, i \in I), \Delta')$ is any family of morphisms $f_i : A_i \to A'_i, i \in I$ (where each morphism f_i belongs to the category D_i) such that for every two points of the scheme $i, j \in I$ and every arrow $s : i \to j$ the following naturality condition holds:

$$F_j(f_j) \circ \Delta(s) = \Delta'(s) \circ F_i(f_i)$$

.

It is easy to verify that such choice of morphisms indeed leads to a category. We will denote it as $\downarrow\downarrow_I F$. It is noteworthy that this category can be obtained using universal constructions in the “category of all categories” $\mathbb{CAT}$:

Theorem 1 *The category $\downarrow\downarrow_I F$ is isomorphic to a vertex of the following pullback in $\mathbb{CAT}$.*

One particular case of this construction is well known in category theory. It occurs when the schema $0 \to 1$, that consists of two points 0, 1, and one nontrivial arrow from 0 to 1, is employed as I. In this case the family F is reduced to a pair of functors

$$F_0 : D_0 \to C \leftarrow D_1 : F_1.$$

The generated category $\downarrow\downarrow_I F$ is known under the name "comma category" [9] and denoted as $F_0 \downarrow F_1$. Having this in mind, we call an arbitrary category of the kind $\downarrow\downarrow_I F$ a *multicomma* category. Theorem 1 allows us to derive a number of properties of the multicomma category which are useful in formal analysis of systems engineering procedures. For instance, if the schema I is discrete (i.e. doesn't contain any nontrivial arrows), then the multicomma category $\downarrow\downarrow_I F$ is isomorphic to a product of categories $\Pi_{i \in I} D_i$ independently on the choice of functors F_i and the aspect C. In other words, all descriptions of a multicomponent product that does not impose any relationship between its components form a conventional Cartesian product of representations of these components, even without having to choose any common aspect. This matches the intuitively clear possibility to place any set of noninteracting things into a common "bag" that is only nominally called the holistic product.

Furthermore, if every functor F_i is an isomorphism (i.e. all components are fully specified in the aspect C), then the multicomma category $\downarrow\downarrow_I F$ is isomorphic to a category C^I that consists of all diagrams of the form I in the category C, i.e. any graph is a valid description of the product. In addition, it is possible to show that the multicomma category construction behaves naturally with regard to sums and products of the schemes of the structure. One can prove a number of other statements that characterize certain particular design decisions on the composition of complex products.

In general, nowadays advances in engineering of applied axiomatic systems fall significantly behind the expectations that arose in the past decades. We hope that the intense employment of methods of machine intelligence and higher algebra will reduce this gap, and ultimately turn the axiomatic method into a powerful tool to solve real-world problems of systems engineering.

References:

[1]. Kovalyov S.P., Rodin A.V. Axiomatic Method in Contemporary Science and Technology: Pragmatic Aspects // Epistemology Philosophy of Science. 2016. Vol. 47, No. 1., p. 153-169. [In Russian]

[2]. Vassilyev S.N., Zherlov A.K., Fedosov E.A., Fedunov B.E. Intellectual Control over Dynamic Systems. Fizmatlit, 2000. [In Russian]

[3]. Muggleton S., De Raedt L. Inductive Logic Programming: Theory and Methods // The Journal of Logic Programming. 1994. Vol. 19-20. P. 629-679.

[4]. Volker J., Haase P., Hitzler P. Learning Expressive Ontologies. In: Ontology Learning and Population: Bridging the Gap between Text and Knowledge. Frontiers in Artificial Intelligence and Applications. 2008. Vol. 167. P. 45-69.

[5]. Sakama C., Inoue K. Can Machines Learn Logics? // 8th International Conference on Artificial General Intelligence (AGI-15). Lecture Notes in Artificial Intelligence. 2015. Vol. 9205. P. 341-351.

[6]. Dai W.-Z., Muggleton S., Zhou Z.-H. Logical Vision: Meta-Interpretive Learning for Simple Geometrical Concepts // 25th International Conference on Inductive Logic Programming (ILP-2015). Kyoto, 2015.

[7]. Szegedy C., Zaremba W., Sutskever I., Bruna J., Erhan D., Goodfellow I.J., Fergus R. Intriguing Properties of Neural Networks. CoRR abs/1312.6199. 2013.

[8]. Graves H. Axiomatic Category Theory for Knowledge Representation // NIST Workshop on Computational Category Theory. NIST, 2015.

[9]. Mac Lane S. Categories for the Working Mathematician. Springer, 1978.

3 Andrei Rodin (Institute of Philosophy RAS) Models of Homotopy Type Theory and the Semantic View of Theories

1. Categorical Model Theory

Today's Categorical Model theory (CMT) stems from the functorial semantics of algebraic theories proposed by Lawvere in his thesis back in 1963 [Lawvere:1963]. This theory uses a family of concepts of model none of which can be called today fairly standard. This fact is evidenced by the continuing discussion in the Homotopy Type theory (HoTT) [UFP:2013] where presently there is no full agreement among the researchers in the field as to what counts as a model of this theory and what does not.

One approch relies on the concept of *classifying category* T freely generated from the syntax of the given theory. Then a model M is a functor $T \to C$ into the category of sets ($C = Set$) or another appropriate category. This functorial setting has an important universal property: up to the categorical equivalence T can be identified with the initial object in the functor category of T-models. This property allows one to think of a theory in this setting as being a "generic model" (Lawvere). Using this approach Awodey [Awodey:2015] defines for HoTT the concept of *natural* model .

Voevodsky [Voevodsky:2015] pursues a different approach, which involves the concept of *contextual category* (more recently - in a modified form of C-system) earlier proposed by Cartmell [Cartmell:1986]. The idea behind the concept of contextual category is that of a category, which fully encodes all relevant algebraic features of the given syntax. According to this approach those and only those categories, which fall under the corresponding definition of contextual category, qualify as models of given theory T. In this case the initiality property of the syntactic category $S(T)$ is not implied by any general theorem. The initiality conjecture for HoTT still stands open.

Finally, there is yet another approac in CMT, which involves the concept of *internal language* (aka *internal logic*) of a given category. It has been recently proposed to think of internal languages and syntactic categories in terms of adjoint functors between a category of theories and a category of categories as shown on the diagram below:

$$Categories \underset{Synt}{\overset{Lang}{\rightleftarrows}} Theories$$

In this setting a model of given theory T in a certain ground category C is a functor (a morphism in the category of theories) of the form

$$M : T \to Lang(C)$$

which expresses the idea of representation of a given theory in the language of some other theory (such as a representation of some geometrical theory in the language of arithmetic).

These and other technical advances of CMT so far have no generally accepted epistemological underpinning, which might help one to orientate among multiple developments. It remains, generally, unclear whether or not the classical Tarskian notion of model based on the T-schema and its standard epistemological understanding can be helpful in CMT. In what follows I show that the classical Tarskian concept of model is not adequate for accounting for the model theory of HoTT in its existing form and propose a remedy. Then I argue that the proposed non-standard understanding of concepts of theory and model can be used for supporting a new version of the semantic of view of theories, which may help to bridge the persisting gap between the notion of model as it is used in logic, on the one hand, and the colloquial notion of model used elsewhere in science, on the other hand.

2. Modeling HoTT

I shall consider HoTT *without* the univalence axiom. In this case the syntax of HoTT is that of (the intensional version of) Martin-Lof's Constructive Type theory (MLTT). HoTT also involves a semi-formal interpretation of its syntax in the Homotopy theory: types are interpreted as spaces (more precisely, infinite-dimensional fundamental groupoids of such spaces) and terms are interpreted as points of these spaces. This interpretation helped to reveal a feature of MLTT's syntax, which earlier remained hidden. Namely, it has been observed that types in MLTT are stratified into the so-called *homotopy levels*. It is important to stress that this stratification is a robust mathematical fact but not just a matter of one's favorite informal interpretation of the given calculus. This stratification necessitates a revision of the informal "propositions-as-types paradigm" , which is popular in the Computer Science. It shows that only types of certain homotopic level (namely, of level (- 1) as defined in [UFP:2013] can be identified with propositions while the higher types should be interpreted differently. This revision implies, in particular, that HoTT cannot be coherently interpreted as a system of propositions or sentences; correspondingly, the Tarskian notion of model based on the T-schema and the satisfaction relation applies only to propositional types (and the corresponding rules) of HoTT but not to this theory as a whole.

MLTT is a system of formal rules without axioms. In the case of propositional types these rules can be called *logical* rules in the usual sense. When these rules are applied to the higher types they should be thought of as rules for constructing non-propositional objects. A *model* of MLTT-HOTT is an implementation of this system of rules in some background, where higher-order constructions play the role of truth-makers for their associated propositions. (A proposition associated with a given higher type T is obtained from T via its (-1)*truncation*). This basic interpretation agrees with all existing models of HoTT disregarding the subtleties mentioned above. An interesting epistemological question is this. Does the epistemic role of higher-order constructions in HoTT reduce to their role as truth-makers or there is something more to it? Since the truncation of higher types to propositional types, generally, involves a significant loss of structure, HoTT rather supports the second answer (unless one assumes that a major part of this theory is epistemically insignificant). In the following concluding section I provide an independent argument, which supports the same conclusion and explains the epistemic value of higher non-propositional structures in HoTT.

3. Semantic View of Theories: a Constructive Perspective

P. Suppes [Suppes: 2002] argued that a typical scientific theory should be identified not with any particular class of statements (formal or contentual) but rather with a certain class of models . On this basis Suppes and his followers designed a Bourbaki-style format of formal presentation where a scientific theory is presented through an appropriate class of itsset-theoretic models.Albeit such a Bourbaki-style presentation can be useful for purposes of logical and structural analysis, it appears to be useless as a practical tool, which may help working scientists to formulate and develop their theories in a formal setting [Halvorson:2015].

Such a limitation is hardly surprising given that the standard set-theoretic semantics of theories provides no formal means for building and operating with models other than by referring to the fact that a model in question satisfies such-and-such propositions. Differences in epistemological views on the roles of syntax and semantics affect the style of formal presentation but not its architecture. This is why in practice the usual *non-statement* aka *semantic* approach to the formalization of scientific theories demonstrates the same limitations as its syntactically oriented rival.

HoTT and its model theory provides a novel notion of theory, which does not reduce to a class of propositions but has a further higher-order non-propositional structure. The axiomatic basis of such a theory consists of a system of rules, which apply both at the propositional and non-propositional levels. I believe that such a broader concept of theory and its model better fits the colloquial counterparts of these notions in the scientific practice than the standard Tarskian notions. The main reason isthat a typical scientific theory involves a lot of *procedural* content, which is used in modeling; such procedures may comprise but typically do not reduce to the procedures of logical inference (if by the logical inference one understands here a procedure which inputs and outputs sentences).

Thus HoTT and CMT provide the semantic view of theories with new formal techniques; the renewed semantic view, in its turn, provides an epistemological background for possible applications of these techniques in science and Knowledge Representation.

References:

[1]. S. Awodey, Natural models of homotopy type theory, arXiv:1406.3219, 2015

[2]. J. Cartmell, Generalised Algebraic Theories and Contextual Categories, Annals of Pure and Applied Logic, 32, p. 209-243, 1986

[3]. H. Halvorson, Scientific Theories, forthcoming in the Oxford Handbook of Philosophy of Science

[4]. F.W. Lawvere, Functorial Semantics of Algebraic Theories. Ph.D., Columbia University, 1963

[5]. P. Suppes, Representation and invariance of scientific structures, Stanford, Calif. : CSLI Publication 2002

[6]. V. Voevodsky, Martin-Lof Identity Types in the C-systems Defined by a Universe Category, arXiv:1505.06446, 2015

[7]. Univalent Foundations Program, Homotopy Type Theory: Univalent Foundations of Mathematics, Institute for Advanced Study (Princeton) 2013; available at http://homotopytypetheory.org/book/,

4 Danya Rogozin (Moscow State University) Curry-Howard Correspondence and Kolmogorov Complexity

1. Preliminaries

1. Decompressor (description method) is a partial function from binary strings to binary strings, $D : \Xi \to \Xi$. If $D(x) = y$ for some $x, y \in \Xi$, than we shall talk that ¡¡x is a description y by D¿¿.

2. By Kolmogorov complexity of binary string by some D we shall mean is the shortest lenght of x, which $D(x) = y$:

$$KS_D(y) = min\{l(x) \mid D(x) = y\} \tag{1}$$

Informally, Kolmogorov complexity is a minimal length of a program that generates current string. If our description method is optimal (the best decompressor), then we talk about Kolmogorov complexity of string itself.

2. Kolmogorov Complexity and Intuitionistic Logic

Let A is a set of binary strings (cardinal number of this set is not important). A will be a task, $x \in A$ will be a solution of this task. Following by the Shen-Vereshchagin method[1] we can define logical operations on this sets of string $(\wedge, \vee, \rightarrow)$:

$$A \wedge B = \{(a, b) \mid a \in A, b \in B\} \tag{2}$$

$$A \vee B = \{(0, a) \mid a \in A\} \cup \{(1, b) \mid b \in B\} \tag{3}$$

$$A \rightarrow B = \{p \mid \forall x \in A\, [p](x) \in B\} \tag{4}$$

Given definitions are based on the next one interpretation of logical connectives. This method goes back to Kolmogorov:

$A \wedge B$ — we can prove A and B.

$A \vee B$ — prove either A or B

$A \supset B$ — proof of B reduces to proof of A.

In the first case, conjunction of tasks has defined as cartesian product of their own desicions. We are going to define Kolmogorov complexity of conjunction as Kolmogorov complexity of pair in general case $(KS(x, y) \leq KS(x) + 2logKS(x) + KS(y) + O(1), x \in A, y \in B)$.

In the second case, disjunction of tasks has defined as union of their desicions. Let $KS(x, y) := min(KS(x), KS(y)) + O(1)$.

In the third case, implication of tasks we will define with a conditional complexity: $KS(x \rightarrow y) := KS(y|x) + O(1)$, $KS(y|x)$ is a complexity of transforming of x to y.

Now we consider some cases:

1. $A \equiv B := (A \rightarrow B) \wedge (B \rightarrow A)$. $KS((x \rightarrow y) \wedge (x \rightarrow y)) = KS(x \equiv y) = max(KS(y|x), KS(x|y)) + O(logKS(x, y))$.

2. $KS((x \wedge y) \rightarrow z) = KS(z|x, y) + O(log(x, y))$. By other hand, $KS(x \rightarrow (y \rightarrow z)) = KS(z|x, y) + O(log(x, y))$.

3. $KS((x \rightarrow z) \wedge (y \rightarrow z)) = max(KS(z|x), KS(z|y)) + O(logKS(x, y, z))$.

[1] *Vereshchagin N. Alexander Shen.* Logical operations and Kolmogorov Complexity. Theoretical Computer Science. Vol. 271(1), 2002. Pp. 125-129

3. Binary λ-Calculus

Lambda-calculus is a formal system (computational model) invited by A.Church in the beginning of 1930-s. Lambda-calculus expresses computational processes by using notions of application and abstraction. Let us define this system formally.

1. Variable x is a term;

2. If M and N are terms then (MN) is term (application rule);

3. If x is a variable and M is a term, then $\lambda x.M$.

At the next step we must introduce De Bruijn notation recursievly by following grammar:

$$e ::= n \mid \lambda.e \mid e\, e \tag{5}$$

Examples ($\Rightarrow$ means ¡¡this term has the next one notation in De Bruijn syntax¿¿):

$\lambda x.x \Rightarrow \lambda 0$

$\lambda x.\lambda y.x \Rightarrow \lambda\lambda 1$

$\lambda f.\lambda g.\lambda x.g(fx) \Rightarrow \lambda\lambda\lambda 1\ (2\ 0)$

$\lambda f.\lambda g.\lambda x.(fx)(gx) \Rightarrow \lambda\lambda\lambda(2\ 0)(1\ 0)$

$\lambda f.(\lambda x.f(xx))(\lambda x.f(xx)) \Rightarrow \lambda(\lambda 1\ (0\ 0))(\lambda 1\ (0\ 0))$

Following by John Tromp[2] we consider the way of binary coding of De Bruijn notated lambda-terms.

1. $n := 1^{n+1}0$;

2. $\widehat{\lambda M} := 00\widehat{M}$;

3. $\widehat{MN} := 01\widehat{M}\widehat{N}$

Let us expand or previous example ($\Longrightarrow$ mean ¡¡this De Bruijn notated term has the next one binary string¿¿):

$\lambda x.x \Rightarrow \lambda 0 \Longrightarrow 0010$

[2] *Tromp J.*, Binary Lambda Calculus and Combinatory Logic, in Randomness And Complexity, from Leibniz To Chaitin, ed. Cristian S. Calude, World Scientific Publishing Company, October 2008.

$\lambda x.\lambda y.x \Rightarrow \lambda\lambda 1 \Longrightarrow 0000110$

$\lambda f.\lambda g.\lambda x.g(fx) \Rightarrow \lambda\lambda\lambda 1\,(2\,0) \Longrightarrow 00000001110011111010$

$\lambda f.\lambda g.\lambda x.(fx)(gx) \Rightarrow \lambda\lambda\lambda(2\,0)(1\,0) \Longrightarrow 0000000101111010011101 0$

$\lambda f.(\lambda x.f(xx))(\lambda x.f(xx)) \Rightarrow \lambda(\lambda 1(00))(\lambda 1(00)) \Longrightarrow 0100000111011101100000111011 10110$

4. Simply Typed λ-Calculus

The most useful form of typed lambda calculus is a calculus with Curry-style typing form. The next one rules are defined for abstaction and application:

$$\frac{\Gamma, x:\phi \vdash M:\psi}{\Gamma \vdash (\lambda x.M):\phi \to \psi}(\to_{In}) \quad \frac{\Gamma \vdash M:\phi \to \psi \;\; \Delta \vdash x:\phi}{\Gamma, \Delta \vdash Mx:\psi}(\to_{El}) \tag{6}$$

Curry-Howard isomorphism[3] is a two-side correlation between computer programs (typed lambda-terms) and proofs in natural deduction style[4], which can be formulated by two principles: ¡¡proof-as-terms¿¿ and ¡¡propositions-as-types¿¿. The first principle claims that lambda-term codes natural-deduction proof, the second one claims that intuitionistically valid proposition corresponds to the inhabited type.

For example, we can prove $\vdash (A \supset B) \supset ((B \supset C) \supset (A \supset C))$ in natural deduction.

1. $A \supset B$;

2. $B \supset C$;

3. A;

4. B — 1, 3, $\supset_{El}$;

5. C — 2, 4, $\supset_{El}$;

6. $A \supset C$ — 3, 5, $\supset_{In}$;

7. $(B \supset C) \supset (A \supset C)$ — 2, 6, $\supset_{In}$;

[3] *Sorensen M.H., Urzyczyn P.* Lectures on the Curry-Howard Isomorphism. — Amsterdam: Elsevier, 2006. Pp. 69 - 102.

[4] *Dummett M.* Elements of Intuitionism. The Second Edition. — Oxford: Oxford University Press, 2000. P. 88.

8. $(A \supset B) \supset ((B \supset C) \supset (A \supset C))$ — 1, 7, $\supset_{In}$.

But we can lambda-term of the next type $(\alpha \to \beta) \to (\beta \to \gamma) \to \alpha \to \gamma$ by the same way:

1. $f : \alpha \to \beta$;
2. $g : \beta \to \gamma$;
3. $x : \alpha$;
4. $fx : \beta$;
5. $g(fx) : \gamma$;
6. $\lambda x.g(fx) : \alpha \to \gamma$;
7. $\lambda g.\lambda x.g(fx) : (\beta \to \gamma) \to \alpha \to \gamma$;
8. $\lambda f.\lambda g \lambda x.g(fx) : ((\alpha \to \beta) \to (\beta \to \gamma) \to \alpha \to \gamma)$

5. Further Research

The goal of a further research is a modifying of the Shen-Vereshchagin method. At the first, it's possible to use type-inference algorithm. Type-inference algorithm returns by given lambda-term its type for polynomial time (this algorithm is applied for type infering in functional languages, Haskell, OcaML, F♯, Idris, etc). That result belongs to Hindley, Milner and Damas. At the second, well-formed (from a type-theoretical point of view) lambda-term has type, which is in conformity with appropriate intuitionistically valid proposition. From a logical point of view, this fact means that we can get a proposition using the code of its own proof.

Without considering all the particulars and somewhat informally, we are going to try to define such a method. At the first step, we represent an arbitrary lambda-term to a binary string following Tromp's rule. Next thing we have to do, we have to get upper bound of Kolmogorov complexity for obtained string. After that, we apply Hindley-Milner algorithm to the given lambda-term. If term is well-formed, then obtained type is the single. Kolmogorov complexity of this type (or proposition, or task) is a Kolmogorov complexity of binary lambda-term. It means that we define complexity of formula (task) by its proof (decision). For example, we have some term $M : \phi \times \psi \to \tau$, then we can define complexity of M as $KS(M) = (z|x, y) + O(log(x, y, z)), z : \tau, x : \phi, y : \psi$ (according to the Shen-Vereshchagin definitions).

BHK-semantics suggests to consider the intuitionistic logic as the calculus of problems (or tasks). Curry-Howard correspondence gives us a formalization of the notion of solution of the task. The kernel of modifying of the Shen-Vereshchagin method is a using binary lambda-terms as solutions of a given problem.

The long-run objective is construction of the general method for defining of complexity of intuitionistic formulas using typed lambda-calculus and Curry-Howard correspondence.

References:

[1]. *Vereshchagin N. Alexander Shen.* Logical operations and Kolmogorov Complexity. Theoretical Computer Science. Vol. 271(1), 2002. Pp. 125-129

[2]. *Tromp J.*, Binary Lambda Calculus and Combinatory Logic, in Randomness And Complexity, from Leibniz To Chaitin, ed. Cristian S. Calude, World Scientific Publishing Company, October 2008.

[3]. *Sorensen M.H., Urzyczyn P.* Lectures on the Curry-Howard Isomorphism. Elsevier, 2006.

[4]. *Dummett M.* Elements of Intuitionism. The Second Edition. Oxford University Press, 2000.

5 Konstantin Shishov (Moscow State University) Reversible Logical Gates in Quantum Logic

1. Preliminaries

According to Moore's thesis, growth of computing power is limited by fundamental physical principles that underlie electronic computing circuits. One of these is the principle of Landauer, according to which the logic circuits, which are not *reversible* should produce heat in proportion to the number of erasure, in the process of computation. Accordingly, the computer scientists are looking for ways to overcome these limitations. One of these ways is introduction of the quantum principles into computational process, which called quantum computing.

However, quantum computing is developing quantum logic - non-classical logic, which involves the construction of logical systems describing quantum computing.

This paper is in line with the quantum logic, and aims to focus on *reversible* nature of logical operations. There exists a close connection between classical reversible computation and quantum computation, since all unitary quantum operations are necessarily reversible; therefore, reversible computing is a subset of quantum computing.

1. Quantum Bits

Consider the two-dimensional Hilbert $\mathbf{C}^2$, where any vector $|\varphi\rangle$ is represented by a pair of complex numbers. Let $B = \{|0\rangle, |1\rangle\}$ be the orthonormal basis for $\mathbf{C}^2$ such that

$$|0\rangle = (0,1); \quad |1\rangle = (1,0)$$

Definition 1. Qubit

A qubit is a unit vector $|\varphi\rangle$ of the space $\mathbf{C}^2$ such that

$$|\varphi\rangle = a_0|0\rangle + a_1|1\rangle,$$

where $a_0, a_1 \in \mathbf{C}^2$ and $|a_0|^2 + |a_1|^2 = 1$.

Futher we will use *x,y,z,...* as a variable ranging over the set {0,1}. At the same time $|x\rangle, |y\rangle, |z\rangle, ...$ will ranging over the basis $\{|0\rangle, |1\rangle\}$. The set of all vectors having the form $|x_1\rangle \otimes ... \otimes |x_n\rangle$, where "$\otimes$" represents the tensor product, represents a computational basis $\otimes^n \mathbf{C}^2$, such as:

$$\otimes^n \mathbf{C}^2 =_{Df} \mathbf{C}^2 \otimes ... \otimes \mathbf{C}^2$$

Now we define n-register.

Definition 2. n-register
A n-register is any unit vector $|\psi\rangle$ in the $\otimes^n \mathbf{C}^2$.

Obviously, the computational basis $\otimes^n \mathbf{C}^2$ can be represented by sequences of $|0\rangle$ and $|1\rangle$ length $n : |101...011\rangle$. Such sequence represents a natural number $j \in [0, 2n-1]$ in binary notation. We obtain that any unit vector of $\otimes^n \mathbf{C}^2$, can be shortly expressed as:

$$\sum_{j=0}^{2^n-1} a_j |j\rangle$$

In accordance with this we will call any vector that is either qubit or an n-register, a quregister.

The model of reversible computation has to fulfil these two conditions: the number of inputs and outputs of the function f has to be the same, and f the model has to have a one-to-one Boolean function. Likewise, we can pose the problem of universality as before, and ask for a set of universal reversible logic gates that can simulate arbitrary reversible Boolean functions.

One of the most famous reversible logical gates is a *Toffoli gate* which is named after its creator and is also known as CCNOT.

Definition 3. Toffoli gate $T^{(1,1,1)}$

The Toffoli gate $T^{(1,1,1)}$ is the linear operator $T^{(1,1,1)} : \otimes^3\mathbf{C}^2 \rightarrow \otimes^3\mathbf{C}^2$ that is defined for any element $|x\rangle \otimes |y\rangle \otimes |z\rangle$ of the basis as follows:

$$T^{(1,1,1)}(|x\rangle \otimes |y\rangle \otimes |z\rangle) = |x\rangle \otimes |y\rangle \otimes |min(x,y) \oplus z\rangle,$$

where "$\oplus$" represents the sum modulo 2.

Obviously, logical gate $T^{(1,1,1)}$ can be interpreted as a simple truth-table that transforms triples of bits (qubits) to triples of bits (qubits).

$$\begin{array}{c}
|0,0,0\rangle \longrightarrow |0,0,0\rangle \\
|0,0,1\rangle \longrightarrow |0,0,1\rangle \\
|0,1,0\rangle \longrightarrow |0,1,0\rangle \\
|0,1,1\rangle \longrightarrow |0,1,1\rangle \\
|1,0,0\rangle \longrightarrow |1,0,0\rangle \\
|1,0,1\rangle \longrightarrow |1,0,1\rangle \\
|1,1,0\rangle \longrightarrow |1,1,1\rangle \\
|1,1,1\rangle \longrightarrow |1,1,0\rangle
\end{array}$$

It is proved that this is an universal logic gate. That is, this can be used to replace all other logical connectives. For example, conjunction:

$$\mathbb{AND}(|\varphi\rangle, |\psi\rangle) =_{Df} T^{(1,1,1)}(|\varphi\rangle \otimes |\psi\rangle \otimes |0\rangle)$$

...and it's truth-table:

$$\begin{array}{l}
(|0\rangle, |0\rangle) \longrightarrow T^{(1,1,1)}(|0\rangle \otimes |0\rangle \otimes |0\rangle) = |0\rangle \otimes |0\rangle \otimes |0\rangle \\
(|0\rangle, |1\rangle) \longrightarrow T^{(1,1,1)}(|0\rangle \otimes |1\rangle \otimes |0\rangle) = |0\rangle \otimes |1\rangle \otimes |0\rangle \\
(|1\rangle, |0\rangle) \longrightarrow T^{(1,1,1)}(|1\rangle \otimes |0\rangle \otimes |0\rangle) = |1\rangle \otimes |0\rangle \otimes |0\rangle \\
(|1\rangle, |1\rangle) \longrightarrow T^{(1,1,1)}(|1\rangle \otimes |1\rangle \otimes |1\rangle) = |1\rangle \otimes |1\rangle \otimes |1\rangle
\end{array}$$

Thus defined, the conjunction will be just as reversible as the Toffoli gate itself is reversible.

All this happens in the simplest situation, when one is only dealing with elements of the basis (in other words, with precise pieces of information). Let us examine the case where the function AND is applied to arguments that are superpositions of the basis-elements in the space $\mathbf{C}^2$. Consider the following qubit pair:

$$|\varphi\rangle = a_0|0\rangle + a_1|1\rangle \text{ and } |\psi\rangle = b_0|0\rangle + b_1|1\rangle\text{,when:}$$

$$\mathbb{AND}(|\varphi\rangle, |\psi\rangle) = (a_1b_1|1,1,1\rangle + a_1b_0|1,0,0\rangle + a_0b_1|0,1,0\rangle + a_0b_0|0,0,0\rangle)$$

Like in a classical logic the function $\mathbb{AND}$ corresponds to the values (1,1,1), (1,0,0), (0,1,0), (0,0,0). But, unlike classical logic, each case is accompanied by a complex number that represents a quantum amplitude - a characteristic reflecting the degree of probability with which the triple can be realized. For example, recording $|a_1b_1|^2$ determines the likelihood that both qubits are unity, and therefore, their conjunction is equal to unity.

Now consider how quantum logic introduces negation. First, it is necessary to consider the function $\mathbb{NOT}$, which is a generalization of the classical negation that negates the value of the last element in the base vector. In this vector notation it looks as follows: if $|x_1, ..., x_n\rangle$ is a vector in the computational basis $\otimes^n\mathbf{C}^2$, the result of the application of $\mathbb{NOT}$ is $|x_1, ..., 1 - x_n\rangle$

In case of denial to one qubit, the function $\mathbb{NOT}$ becomes a single function that associates elements-arguments from the space $\mathbf{C}^2$ elements, the values of $\mathbf{C}^2$.

$$\mathbb{NOT}^{(1)} =_{Df} (a_1|0\rangle + a_0|1\rangle)$$

It may be noted that the thus defined function $\mathbb{NOT}$ is a generalization of the classical negation.

In general, the function $\mathbb{NOT}$ is defined as follows:

$$\mathbb{NOT}^{(n)} : \otimes^n\mathbf{C}^2 \longrightarrow \otimes^n\mathbf{C}^2 \text{ and for all } |\varphi\rangle = \sum_{j=0}^{2^n-1} a_j|j\rangle \in \otimes^n\mathbf{C}^2$$

$$\mathbb{NOT}(|\varphi\rangle) = \sum_{j=0}^{2^n-1} a_j|x_{j_1}, ..., x_{j_{n-1}}, 1 - x_n\rangle$$

3. Quantum Logical Gates as Reversible Gates

In quantum computing and specifically the quantum circuit model of computation, a quantum logic gate is a basic quantum circuit operating on a small number of qubits. They are the building blocks of quantum circuits, like classical logic gates are for conventional digital circuits. Unlike many classical logic gates, quantum logic gates are reversible. However, it is possible to perform classical computing using only reversible gates. For example, the reversible Toffoli gate can implement all Boolean functions. This gate has a direct quantum equivalent, showing that quantum circuits can perform all operations performed by classical circuits.

The logical gates we have considered so far are, in a sense, generalizations of the classical logical connectives. A quantum logical behaviour only emerges in the case where our gates are applied to superpositions. When restricted to classical registers, our gates behave like classical truth-functions. We will now investigate genuine quantum gates that may transform classical registers into quregisters that are in superpositions.

Definition 4. Quantum logical gate

Quantum logical gate is an unitary operator, assuming arguments in a $\otimes^n \mathbf{C}^2$ values in $\otimes^n \mathbf{C}^2$.

One of the most significant genuine quantum gates is the square root of the negation (NOT), which will be indicated by $\sqrt{\mathbb{NOT}}$. As suggested by the name, the characteristic property of the gate $\sqrt{\mathbb{NOT}}$ is the following: for any quregister $(|\psi\rangle)$

$$\sqrt{\mathbb{NOT}}(\sqrt{\mathbb{NOT}}(|\psi\rangle)) = \mathbb{NOT}(|\psi\rangle)$$

A more general definition of $\sqrt{\mathbb{NOT}}$ is as follows:

$$\sqrt{\mathbb{NOT}}^{(1)} : \mathbf{C}^2 \longrightarrow \mathbf{C}^2 \text{ and for all } |\psi\rangle = a_1|0\rangle + a_0|1\rangle$$
$$\sqrt{\mathbb{NOT}}^{(1)}(|\psi\rangle) = \tfrac{1}{2}[(1+i)a_0 + (1-i)a_1]|0\rangle + \tfrac{1}{2}[(1-i)a_0 + (1+i)a_1]|1\rangle,$$

where i is an imaginary unit.

So $\sqrt{\mathbb{NOT}}^{(1)}$ translate two basic-station of qubit $|0\rangle$ and $|1\rangle$ into the superposition of this states.

Consider another "purely" quantum gate:

Definition 5. Hadamard-gate (H-gate)

For all qubit $n \geqslant 1$ Hadamard-gate $\otimes^n \mathbf{C}^2$ is a linear operator $H^{(2^n)}$ such as $|x_1, ..., x_n\rangle$ computational basis $\mathcal{B}^{(2^n)}$:

$$H^{(2^n)}(|x_1, ..., x_n\rangle) = |x_1, ..., x_{n-1}\rangle \otimes 1\sqrt{2}((-1)^{x_n}|x_n\rangle + |1 - x_n\rangle$$

Thus it turns out that for $n = 1$ the use of the Hadamard-gate will have the form $H^{(2^n)}(|x\rangle) = H(|x\rangle)$ own case, while, as in the $n > 1$, we can see its connection with other logical gate: $n > 1$ $H^{(2^n)}(|x_1, ..., x_n\rangle) = I^{n-1}(|x_1, ..., x_{n-1}\rangle) \otimes H(|x_n\rangle)$

Hadamard-gate mapping $|0\rangle$-basis into $\frac{|0\rangle+|1\rangle}{\sqrt{2}}$, and $|1\rangle$-basis in $\frac{|0\rangle-|1\rangle}{\sqrt{2}}$, that corresponds to a rotation about the axis π. A double application Hadamard-gate will correspond to *identity gate*, which maps its input to the output unchanged.

4. Further Research.

This investigation presents an overview of some aspects of the quantum and reversible computation. It serves as a better understanding of the specific characteristics of quantum logic. In addition, in the process of studying the material we singled out some unresolved issues that form the scientific foundation for the future research. One of such problems is to compare unitary operations in quantum logic with modal operators. Is it possible to convert the work of processes in quantum logic to work with modal operators? Or, more broadly, is it possible to reduce quantum logic to modal logic?

References:

[1]. Cattaneo G., Dalla Chiara M.L., Giuntini R., Leporini R. *An unsharp logic from quantum computation* // International Journal of Theoretical Physics, 2004, Volume 43, Issue 7-8, pp 1803-1817

[2]. Dalla Chiara M.L., Giuntini R., Greechie R. *Reasoning in Quantum Theory*, Springer, 2004

[3]. Dalla Chiara M.L., Giuntini R. *Quantum logic* // Handbook of Philosophical Logic, Vol. 6, Kluwer Academic Publishers

6 Sergei Titov (Institute of Philosophy RAS) Data Science and Scientific Explanation

1. Issue

For the last years amount of available data of any kind for research grew up significantly, especially in domain of social sciences [Sagiroglu & Sinanc: 2013]. One can easily buy any amount of public data collected by such companies like GNIP or participate in data grant programs. Today we overcame lack of data about surrounding world - probably every human action is somehow digitalized and saved, every nature event is carefully documented. Key problem now is to decide what to do with all this data - incompetent research may lead us to absurd results [Bennett et al.: 2010] at best case. It may seem that such problem statement is speculative, but in this work we will try to show how data may affect the study.

2. Data and Phenomena

Woodward and Bogen argue about relations between data, phenomena and theory [Bogen & Woodward: 1988]. Main idea of the work is that phenomena is something that exists but is unobservable while data plays a role of evidence of the existence of such phenomena. In other words, Bogen and Woodward indue phenomena with ontological status.

Point here is that such phenomena is strictly unobservable as example authors demonstrate that regardless of technology we use it is impossible to know true value of melting point of lead. In case of the melting point we always end the research with some database of temperatures and we need to make decision about final value. From the Woodward and Bogen point of view this decision in a way is irrelevant because phenomena is in laws of nature" domain and some true value is actually in the data.

In a critique on Woodward and Bogen paper, MacAllister [McAllister: 1997] pays closer attention on this decion making. He claims that this decision is scientist personal choice so phenomena doesn't actually have ontological status. In this case deducing phenomena from a data become psychological issue - way of scientist observe data become essential for phenomena detection.

Longo and Calude [Calude & Longo: 2015] present formal restrictions in phenomena detection. They build argument upon "spurious correlations" which is based on Ramsey theory. Here we will briefly describe their argument.

Ramsey theory suggest that for every predefined structure there is big enough set where it can be guaranteedly found. For example. A special case is given by the following theorem due to van der Waerden: For any positive integers k and c there is a positive integer γ such that every string, made out of c digits or colors, of length more than γ contains an arithmetic progression with k occurrences of the same digit or color, i.e. a monochromatic arithmetic progression of length k [Shkredov:2006].

In case of big enough database we can find there arbitrary long progression of correlations in order to consider them relevant? Here contradiction emerge - reliability of correlation may be caused only by size of database - we devoted to found any correlations in big enough databases.

Longo and Calude define spurious correlations as a “correlation is spurious if it appears in a 'random' generated database”. Despite strictness of this definition it allows us to estimate number of such correlation using Kolmogorov theory. In the paper authors bring out calculations for correlations spuriousness” for 2048 symbols long string and get in result that overwhelming majority of possible correlations could be considered as spurious.

An interesting fact is that Kolmogorov theory and Ramsey theory in case of this argument have a intriguing contradiction: if we will take a long enough and completely random string (in Kolmogorov sense) it would have some predefined pattern according to Van der Waerden theorem. Can we consider such string truly random?

While this paper stops just on this statement there is one more step to be done: what to do with spurious correlations? This particular paper is not focused on such question and poof above may allow us only to argue about of impossibility of replacement science with data analysis.

3. Embracing Descriptive Science

Another view on data-intensive science lies beyond the problem of phenomena detection. If we agree that all patterns in data are some phenomenas we would run into another epistemological problem.

In his recent paper Pietsch suggest to distinguish vertical and horizontal type of science. First one is classical view on science where researches have some structure of concepts - from the most universal laws to practical and especial cases. These concepts are consistent and can be deduced from each other. Good example of vertical science structure is physics but it is the exception rather than the rule - the most of the knowledge areas doesn't fit in such standards, so there is horizontal science.

Horizontal model describe disciplines like social sciences or economics in which vertical model has failed. While vertical model rely on reduction and theoretical strictness horizontal model based on efficient analysis of big amount of data. Pietsch outline four main properties of horizontal model:

- Predictions are made from the data directly without casual structure of phenomena

- There is no need in high-abrstrat concepts because probably every aspect of the phenomena is already in data
- Lack of explanatory power in models
- Rare use of idealisations and simplifications

Pietsch [Pietsch:2015] suggests that large enough database contain every state of phenomena which change probabilistic inference into unambiguous one. Such procedure we can consider as a case of eliminative induction and transition from descriptive statistics to causal inference. So complete horizontal model will provide casual predictions like vertical does but with serious lack of explanatory power.

Napoleani and Panza [Napoleani & Panza:2011] provide similar view on future of special sciences (in Woodward sense). Like Pietsch they argue that deductive model have failed in this type of disciplines and suggest to concentrate scientific attention specifically on data analysis. They propose term of "agnostic science" which imply the idea of "turn over" relation of mathematical methods and empirical research. In classical science mathematical apparatus was usually developed specifically for every phenomena, but in "agnostic" science Napoleatani and Panza argue that research would seek already developed mathematical model for data which describes phenomena:

"Mathematics becomes perhaps the only domain in which to develop structural understanding, since such pretense is lost in the study of phenomena. Ideas are then forced upon the phenomenon in problem solving, only temporary, and with little expectations that go further than the solution of the problem. Scientific methods may become weak, but the mathematical language in which they are phrased will be increasingly complex, as we attempt to mould our desires, coarsely, upon reality."

Good example for such idea is hurricane prediction by neural networks. Despite the fact that we have physical model for this, predictions made by neural network with base only on data are much more precise. But the price of such efficiency is explanatory power. Despite of examples in the paper this approach begs the questions of possibility of further progress of the phenomenon research after applying independent developed structure.

4. Conclusion

We have considered two interrelated problems. The first problem appears in phenomena discovery in data - in large enough databases we could find any pre-defined pattern because only of the size of database. It make unreliable some of recent scientific findings - for example, false-positive results of fMRI - analysis [Eklund & Nichols & Knutsson : 2016]. The second problem appears in case of complex data analysis. There is a wide variety of methods which rely on model generation of data like neural networks or boosting algorithms. Even if one obtains fascinating results one looses an explanation of the studied phenomena.

In case of the first problem further work would be concentrated around Longo and Caulde argument - primary task is to check some known datasets for spurious correlation probability. In addition there are some ways to strengthen the argue, for example with use of more powerful theorems, like Semeredy theorem [Shkredov:2009].

The second problem raises the question of explanation power loss which could be very perspective from the philosophy of science point of view. Development of the new methodologies which could help to conduct explanatory research in data intensive science is essential for the science.

References:

[1]. Bennett C.M. et. al. Neural Correlates of Interspecies Perspective Taking in the Post-Mortem Atlantic Salmon: An Argument For Proper Multiple Comparisons Correction // J. Serendipitous Unexpected Results. 2010. . 1. 1. . 15.

[2]. Bogen J., Woodward J.F. Saving the Phenomena // Philos. Rev. 1988. . 97. 3. . 303352.

[3]. Calude C., Longo G. The Deluge of Spurious Correlations in Big // Found. Sci. 2015. . 1.

[4]. Eklund A., Nichols T.E., Knutsson H. Cluster failure: Why fMRI inferences for spatial extent have inflated false-positive rates // Proc. Natl. Acad. Sci. 2016. . 16.

[5]. McAllister J.W. Phenomena and patterns in data sets // Erkenntnis. 1997. . 47. 2. . 217-228.

[6]. Napoletani D., Panza M. Agnostic Science . Towards a Philosophy of Data Analysis // Found. Sci. 2011. 16. . 1-20.

[7]. Pietsch W. The Causal Nature of Modeling with Big Data // Philos. Technol. 2015. . 1-37.

[8]. Sagiroglu S., Sinanc D. Big data: A review // 2013 International Conference on Collaboration Technologies and Systems (CTS). , 2013. . 42-47.

[9]. Shkredov I.D. Semeredi Theorem and Problems of Arithmetical Progressions // Uspekhi Matematicheskikh Nauk 2006 v. 6, N 372, p. 111-179 [in Russian].