Cognitive Systems Research 83 (2024) 101175

Contents lists available at ScienceDirect

Cognitive Systems Research

journal homepage: www.elsevier.com/locate/cogsys

Computationally inspired cognitive modeling

Viacheslav Wolfengagen [a,*], Larisa Ismailova [a], Sergey Kosikov [b]

[a] National Research Nuclear University "Moscow Engineering Physics Institute", Moscow 115409, Russian Federation
[b] NAO "JurInfoR", Moscow 119435, Russian Federation

ARTICLE INFO

Action editor: Alexei V. Samsonovich

Keywords:
AGI
Cognitive modeling
Information process
Channeling
Computational thinking
Semantic information processing
Computational model
Variable domains

ABSTRACT

A computational approach to cognitive modeling is proposed. The computational model is a parametric construction that takes into account cognitive stages and transitions between them. The cognitive model enables the idea of information processes, from their birth and appearance in a scope, evolution and canceling out their existence and disappearing from the scope. Process habitats are Lawvere's variable domains; inter-transition is based on the notion of channeled spreading of processes.

1. Introduction

The current explosion in cyberattacks in the form of malware, spam and intrusion has had dire consequences for society. For organizations and government structures, the main concern has become the security of cyberspace. Traditional machine learning (ML) methods are widely used in cyber threat detection, but they are unlikely to model correlations between real cyber entities, which exhibit high variability. In recent years, graph analysis methods, including knowledge graphs, have become widespread, and many researchers have begun to study and develop these methods. The purpose of graph analysis is to identify links between cyber entities, establish correlations between them and achieve high performance. Graph-based methods for solving cybersecurity problems that exist today need to be improved not only in details, but also in their very essence, if we care about their future. Conceptual mathematics (Lawvere & Schanuel, 1997; Marquis, 2015; Schubert, 1972) is increasingly asserting itself, showing its effectiveness.

On this basis, as a key contribution to the problem, we will present an approach and providing a model structure for *graph mining*, suitable for the needs of cognitive modeling and cybersecurity. These include *entanglement* problems, typical *conceptual dependency* analysis methods, and a general process for applying them based on the systematic use of *computational thinking* principles. For each task, we study the appropriate methods and highlight the types of graphs, approaches to graphs and levels of tasks in their modeling.

The basics of graph mining are laid out in Gyöngyi, Garcia-Molina, and Pedersen (2004), aspects of graph analysis are explored in Yan et al. (2023). In today's world, cybersecurity is becoming fundamental, and one cannot ignore the success of applying mathematics to the analysis of processes in the natural sciences (Bangu, 2016; Wigner, 1960). That this manifests itself is no longer disputed, and logic has become an everyday cognitive tool in computer science (Halpern et al., 2001). The observed ubiquity of computing in all spheres of life and activity of the modern world increases the cognitive load on conceptual mathematics (Lawvere & Schanuel, 1997) and, accordingly, computational thinking (Denning & Tedre, 2021).

When it comes to cognitive modeling and cognitive maps, as well as semantic networks, there are two basic and well-known questions in the cognitive sciences to answer: 'What's in a link?' by William A. Woods, asked in 1975 (Woods, 1975) and Ronald J. Brachman's 'What's in a Concept' or, equivalently, 'What's in a node' given in 1977 (Brachman, 1977). To answer these questions, the path of logic was chosen, which was followed by the vast majority. Only a few, including David Spivak in 2014 (Spivak, 2014), began to look for a way to conceptual mathematics. In fact, the reasons for this state of affairs remain unclear.

The content of the work is reflected in the main sections. Section 2 discusses the features of related collective models. Section 3 briefly discusses the relationship between conceptual mathematics, computational thinking, and cognitive modeling. Section 4 introduces the notion

* Corresponding author.
E-mail addresses: jir.vew@gmail.com (V. Wolfengagen), jir.lyui@gmail.com (L. Ismailova), kosikov.s.v@gmail.com (S. Kosikov).
URL: https://wolfengagen.mephi.ru (V. Wolfengagen).

https://doi.org/10.1016/j.cogsys.2023.101175
Received 26 May 2023; Received in revised form 19 July 2023; Accepted 18 September 2023
Available online 21 September 2023

of a cognitive stage. Section 5 denounces the conditions for providing referentiality and referential clarity. Section 6 discusses the transition of the process to a later cognitive stage. Section 7 captures 'actual' world in its generic cognitive stage. Section 8 considers the transition of process domains to a possible 'later' cognitive stage, given the evolvent 'along' which events evolve and the process property changes. Section 9 contains the development of the notion of channeled process propagation and the idea of constructing valid channel closures. This is somehow due to the conceptual composability of channels. Section 10 contains a brief discussion. Section 11 draws the main conclusions and discusses the properties of the generated cognitive model.

2. Related works and models

The dynamics of the self-pattern is considered in Hölken, Kugele, Newen, and Franklin (2023). Here, they focused on the interdependence of narrative and embodied aspects of the self-pattern, since they involve particularly salient challenges There are fruitful ideas in conceptualizing the interaction between propositional and motor knowledge representations.

The model of learning is studied in Kugele and Franklin (2021). It is based on a kind of conceptual commitments that constrain the learning mechanisms. Despite the achieved successes, many conceptual challenges remain yet uncovered. The authors explore foundational issues in learning, such as, 'What must be innate or built-in?' vs. 'What can be learned?', this leads to the study of relationship between the LIDA conceptual model and its computational realizations.

The premise annotation in mental model construction in Brüssow, Ragni, Frorath, Konieczny, and Fangmeier (2013) led to the ability for multiple interpretations that requires maintaining intermediate representations that, if necessary, may be reconsidered at a later stage of the solution process.

Deconstructing and reconstructing in Stewart and West (2007) is explored for the architectural space. The authors argue that evaluating variations in the structure of computational models of cognition is as important as evaluating variations in the numerical parameters of such models.

An interesting attempt of inferring cognitive models from data using approximate Bayesian computation is given in Kangasrääsiö et al. (2017). They tried to uncover important problem for HCI researchers how to estimate the parameter values of a cognitive model from behavioral data. The case study results demonstrate that approximate Bayesian computation (ABC) (i) improves estimates of model parameter values, (ii) enables meaningful comparisons between model variants, and (iii) supports fitting models to individual users. The continuation is in Kangasrääsiö, Jokinen, Oulasvirta, Howes, and Kaski (2019) with study of parameter inference for computational cognitive models with approximate Bayesian computation.

An approach to investigate the potential of logic programming (LP) to computationally model morality aspects studied in part in Saptawijaya and Pereira (2015). The LP serves as the basis for a whole system, into which other reasoning facets will be integrated, to model the surmised morality aspects, and this can give the sound computational ground.

An interpolation approach for fitting computationally intensive models is studied in Richard Moore and Gunzelmann (2014). Computational cognitive modeling is assumed as a useful methodology for exploring and validating quantitative theories about human cognitive processing and behavior. It is noted that complex models can create challenges for parameter exploration and estimation due to extended execution times and limited computing capacity. That paper gives a methodology and associated software for further experiments that allows modelers to instantiate a model proxy that can quickly interpolate predictions of model performance anywhere within a defined parameter space.

An adaptive temporal-causal network model to analyze extinction of communication over time is covered in Johannes José Fijen, Joaquín López González, and Treur (2021). As known, the persistence of information communicated between humans is difficult to measure as it is affected by many features. Hence a variety of computational models should and can be proposed. It is argued that the adaptive mental network model explains, for example, how an individual can experience information overflow on a topic, and how this affects the sharing of information. The parameter tuning should fit to given and taken empirical data.

In Ismailova, Wolfengagen, and Kosikov (2022d), the basic applicative system, whether it is a lambda calculus or a system of combinators, is considered as a concept prototype system, with which we can build applied systems that are practically significant for mathematics, computing, or programming. These are families of computational models that have both their own semantics and application areas. This conceptualization/individuation technique is a characteristic of the field of semantic studies. As it turns out, the applicative approach forms a metatheoretical framework that provides the basis for cognitive systems that consider abstract objects and interpret their properties and behavior in the modern computing environment.

In Ismailova, Wolfengagen, Kosikov, and Andronov (2023) it was shown that the use of a composition of objects representing a data structure, in fact, means the creation on their basis of a kind of information channels through which the computational process is distributed. In the case of applying applicative computing technology, the calculation process is launched by the operation of applying a function object to an argument object. Two ways of generating function objects are presented — use either lambda expressions or combinators to represent them. The first method uses an abstraction meta operator and associated variables, which leads to the use of substitution systems with potential side effects. In the second method, only the constant combinator objects participate in the construction of a function, and applying them to an argument triggers a conversion, based on rewriting rules, which does not cause a side effect. A practical solution to the problem of synthesizing a compositional data structure can be mixed, when both lambda terms and combinators are involved in the calculations, which reduces the length of expressions. As a result, a data structure appears, consisting of compositions of argument objects equipped with a generated set of supporting function objects.

It is noted in Wolfengagen, Ismailova, and Kosikov (2022) that computational activity is now recognized as a natural science, and computational and information processes are found in the deep structures of many areas. Computing in the natural world existed long before the invention of computers, but a noticeable shift in understanding of their fundamental nature is actually taking place before our very eyes. The present moment, in fact, is the transition from the idea of computer science as an artificial science to the understanding that information processes abound in nature. Informatics is recognized as a natural science that studies natural and artificial information processes. In everyday computing, operations are performed on individual generators with little attention paid to their internal structure. However, many common operations consist of more primitive constructs linked by the combination mode. The interaction of information processes and the corresponding structures is carried out in the environment of 'applicative interaction', their applications to each other, and the study of the properties of this environment allows us to understand the nature of the computations. In this paper, the main attention is paid to the clarification of the technological features of calculations with individual generators or objects. Their interaction is considered in the applicative environment, which allows us to find out the internal structure of ordinary operations, the knowledge of which allows us to understand their properties. The choice of generators of initial constants, considered as general and expressed by combinators, is discussed. These source generators are used as the basic 'building blocks' that emerge within the larger applicative environment blocks when interacting with each other. As a result of the interaction, constructions arise that give representative sets of ordinary operators and embedded computing systems.

A semantic metalanguage (Wolfengagen, Ismailova, Kosikov, & Babushkin, 2021) was developed to study the emergence, propagation, and safe interaction of semantic processes in information modeling systems, including cognitive interference. An approach to building a semantic network is proposed, based on a computational model in which both nodes and arcs are informational processes. Concepts are represented by intensional objects within untyped theories, and they, in turn, are considered as special analogues of typed theories. Similar mixing has been used in model studies for lambda calculus. In contrast, in this work, information processes correspond to parameterized metadata objects, which are variable scope constructs. Transformations of variable domains correspond to the spread of the process. A directed transformation provides the generation of metadata targets as parameterized concepts. This models the development of the process, which corresponds to the propagation of cognitive interference and allows us to interpret the hidden time factor. The emerging model is solely based on processes and provides such a conceptual framework. The possibility of coding this framework with a system of interdependent lambda terms is reflected.

The task of modeling a subject whose cognitive activity is directed under the influence of the information environment continues to be relevant (Ismailova, Wolfengagen, & Kosikov, 2022e). Difficulties in solving this problem are due to (1) the dependence of the truth of information on time, as a result of which the subject can make decisions based on outdated information, and (2) the heterogeneous nature of the information itself, which, in particular, can be deliberately false (false). The paper proposes a model of interaction between subjects, including the exchange of information, marked both by the time of its creation (receipt) and the quality of information. The model is based on a system of interacting processes and is presented as a set of information graphs that describe the stages of interaction. A typification system for graphs and their constituent elements is proposed. Composition tools are presented that allow you to assemble complex graphs from simpler ones. The paper also presents a set of tools corresponding to the proposed model. These tools include a process language interpreter that computes model constructs, as well as an information graph editor, type management tools, debugging tools, and configuration management tools for the computing environment. The proposed tools have been experimentally tested in the development of applied systems from the field of implementation of the best available technologies.

In Ismailova, Wolfengagen, and Kosikov (2022a) is discussed the use of conceptual mathematics to construct a computational model of concepts and individuals. The applicative approach is systematically applied to construct the concept as a process. Since modern computing considers information processes as the main objects of modeling, the developed mechanism really represents the semantic processing of information. The nature of concepts – what concepts are – and the constraints that govern the theory of concepts have been, and continue to be, a matter of debate. It is especially interesting to discuss the nature of concepts in connection with the recently established fundamental nature of information processes inherent in all phenomena and events occurring in the world around us. The current trend is elevating information processes into forms of computation that can also be implemented through practices, such as programming. The deep component is computational models, one way or another expressed by means of mathematics and metamathematics. The main meta-operations used are abstraction and application. Of greatest interest is functional abstraction and application in the form of applying a function to an argument. Despite this 'conceptual minimalism', a rich theory of concepts can be developed. Using this theory, it is possible to focus further discussion not only on the nature of concepts, but also to characterize the position on each of the five important issues that are central to many theories of concepts: (1) the ontology of concepts, (2) the structure of concepts, (3) empiricism and nativism in relation to concepts, (4) concepts and natural language and (5) concepts and conceptual analysis.

On the other hand, Ismailova, Wolfengagen, and Kosikov (2022c) discusses Applicative Computing Technology (ACT), which performs semantic analysis of a number of natural language constructs. The necessary elements of grammatical analysis are involved. Many first-order logical tools are used, and predicate variables are necessarily used for analysis. At the same time, the advantages of higher-order systems, to which ACT belongs, are extracted. The fact is that the semantics of a natural language is characterized by a multilevel nesting of grammatical structures, which contradicts first-order logical systems.

The problem of generating words in a context-sensitive language is covered in Slieptsov, Wolfengagen, and Kosikov (2022). It deals with the application task of unit testing of functions, discusses in details the use of generators in property-based testing. A context-sensitive λ-calculus language with a simple type system is considered, and the problem is refined up to the generation of λ-terms of a given type in a given context. A generation method is proposed, implemented using combinators of the fast-check library, which provides property-based testing in JavaScript.

A notion of a 'conceptual hanger' (Ismailova, Wolfengagen, & Kosikov, 2022b) is introduced at a qualitative level, where we mean a concept as an abstraction based on the characteristics of perceived reality. In analytic activity, the concept is considered as some kind of marking imposed on the phenomenon, which forces us to connect individual observations and make generalizations. For convenience, it is identified with the name given to observations and events. Each concept is associated with some of its properties. A variable is often understood as a measured concept as a property that is associated with the concept and changes during measurement. If the property does not change, then it is considered constant. We proceed from the fact that concepts should have a mechanism for change. Under these assumptions, a two-parameter commutative diagram is constructed that connects possible worlds and property transformations. The resulting semantic map sets the basis for their variation. Thus, the regulation of the variability of the associated structure of variable concepts is achieved, which determines the possible relations of movement of concepts.

3. CT vs. AGI: Computational Thinking and Artificial General Intelligence

3.1. Computational thinking

The concept of thinking machines (Hughes & Hughes, 2019; Vallverd & Vallverd, 2010) has always served as a benchmark for computing. LISP machines (Goto et al., 1982) played a special role as a 'knowledge assembler'. The modern definition of computational thinking, according to P. Denning and M. Tedre (Peter J. Denning, Matti Tedre) (Denning & Tedre, 2019), is as follows.

Computational thinking is defined as the process of identifying a clear, defined, step-by-step solution to a complex problem. Its definition includes breaking down a problem into smaller parts, recognizing patterns, and removing extraneous details so that a step-by-step solution can be replicated by humans or computers.

This problem solving model is used in our daily lives not only in computer science, but also in language, history, science, mathematics and art. While there is such a thing as 'disconnected' computational thinking, modern computational thinking often includes a solution containing technology, such as a computer, to execute an algorithm.

Four parts of Computational Thinking are determined now. Wether computational thinking is used in computer science or another subject area, the process of computational thinking can be broken down into four parts or steps.

1. Decomposition

The first step in computational thinking is decomposition. While it may go by different names depending on the school of thought, the basic process is the same: to solve a complex problem, you must first break it down into smaller, more manageable pieces.

Decomposition is an important part of computational thinking because it helps make the problem more manageable (you have probably heard the expression that 'the best way to eat an elephant is one bite at a time'). It also helps problem solvers better define and understand the problem being solved, allowing them to simplify the problem through pattern recognition and abstraction.

2. Pattern recognition

Part of computational thinking is also pattern recognition. It is the process of identifying patterns or connections between different parts of a larger problem. The goal of pattern recognition is to simplify the problem even further by discovering where details may be similar or different, as well as creating a continuous understanding of a more complex problem.

3. Abstraction

Abstraction is the process of extracting the most relevant information from each component task. It helps to define or summarize what exactly needs to be done to solve the problem as a whole. This step in the computational thinking process helps us determine how these important details can be used to solve the same problem from other fields.

4. Algorithmic thinking

The final component of computational thinking is algorithmic thinking. It is the process of determining a step-by-step solution to a problem that can be replicated for a predictable and reliable outcome. For the modern definition of computational thinking related to computer science, this decision would be a step-by-step process to be completed by a computer. However, this process can also be completed partially or completely by humans.

3.2. AGI — artificial general intelligence

According to Nicholas Cassimatis (Cassimatis, 2012), the problems of artificial intelligence and cognitive modeling have similar difficulties. Questions are raised in the field of artificial general intelligence (AGI), which are theoretically, conceptually and philosophically important for the creation of thinking machines. As it turns out, it is *cognitive modeling* that is the challenge at the present stage. Without a definite and serious shift in this direction, the development of all other types of AI is also hampered.

For these reasons, understanding how the human brain embodies the solution to the problem of human intelligence is an important goal of cognitive science. At least at first glance, they are very far from achieving this goal. There are no cognitive models that could, for example, fully understand language or solve simple tasks for a small child. If we evaluate the prospects for the application of existing methods and standards in cognitive science to solve this problem, it is ultimately suggested that it is useful even to create a new sub-field in cognitive science called the *science of intelligence*, but first of all, at least in general terms, some guidelines for this field will have to be outlined.

Before discussing how effective the methods and standards of cognitive science are in solving the problem of intelligence, it is useful to list some of the problems or questions that the science of intelligence must answer.

Usually they build a sequence or hierarchy of AIs that are not identified.

How can we determine, and what is the difference with other AI? Artificial General Intelligence AGI can be defined as a synthetic intelligence that works in a wide range of tasks and has a good ability to generalize in different contexts while performing heterogeneous tasks. In other words, an AI capable of performing assigned tasks as well as a human.

The AI hierarchy at the moment can be represented as follows (ranking from simpler to more complex):

Artificial Narrow Intelligence (ANI, Narrow AI) — specializes in one area, solves one problem.

Artificial General Intelligence (AGI, Strong AI) — capable of performing most of the tasks that a person is capable of.

Artificial Super Intelligence (ASI) — surpasses the capabilities of the intellect of any of the people, is able to solve complex problems instantly.

The definition of AGI can be illustrated by comparing Narrow AI (ANI) and Strong AI (AGI). It is worth noting that now there is not a single system that could be called Strong AI with confidence — all that we now see is AI systems that succeed in performing narrow tasks: detection, recognition, translation from one language to another, image generation, text generation, but so far it is difficult to talk about any universality of such models and the applicability of one model to perform tasks that differ significantly from each other without the need for additional training. It is the applicability of a single AI to perform the above tasks that is most often called strong artificial intelligence.

Although AGI differs from conventional AI in its holistic attitude towards intelligence, the design and development of an AGI system still has to be done step by step, and some of the topics covered are considered more important and substantive than others. This also applies to cognitive modeling.

Cognitive Modeling is necessary to build an understanding of human-level intelligence. Ultimately, we cannot say with certainty that we have understood how the human brain embodies the solution to the problem of intelligence if we do not have: (1) a *computational model* that behaves as intelligently as a human, and (2) some way to know that the *mechanisms* of this model, or at least its behavior, reflect what happens in humans.

Creating computer models that behave like humans and demonstrating that the mechanisms of the model correspond at some level to the mechanism that underlies human cognition is much of what most cognitive modelers are striving for today. Thus, understanding how the human brain embodies the solution to the problem of intelligence is partly a problem of cognitive modeling.

Let us return again to cognitive modeling and introduce some certain systemic nature in its application. Cognitive modeling is understood (nevertheless!) as a field of computer science that deals with modeling problem solving and human mental processing in a *computational model*. Such a model is considered an attribute of *computational thinking*. Here the temptation arises, is it not worth accepting the thesis — cognitive modeling through computational thinking? Such a model can be used to simulate or predict human behavior or performance when performing tasks similar to those being modeled and to improve human–computer interaction.

In fact, computational thinking has been brought into the realm of AI since its very beginnings (Turing, 1950), (McCarthy, Minsky, Rochester, & Shannon, 1955). Even in this direction, large-scale efforts have been made (Feigenbaum & McCorduck, 1983; McCarthy, 1978; Weinreb & Moon, 1981). Accumulated and analyzed the experience gained (Brachman, 2006), difficulties and obstacles were noted.

3.3. Information processes and conceptual mathematics

The well-known thesis of Galileo (Galileo Galilei), that the book of nature is written in the language of mathematics, still works in many respects. Later, Brouwer's intuitionism rejected the uniqueness of such a description. Intuitionism is based on the idea that mathematics is a creation of the mind. The truth of a mathematical statement can only be grasped by a mental construct that proves its truth, and communication between mathematicians is only a means of creating the same thought process in different minds. The subject of mathematics are systems of mathematical objects. In this case, a system is understood as a set of objects with a set of relationships that exist between these objects. Mathematical objects are abstract idealized objects. Mathematical objects play an important role in the formation of mathematical theories.

An abstract object is an object endowed with the properties contained in its definition. Mathematics explores forms and relations that are completely abstracted from content, preserving in them only what is contained in their definitions. Now proposals are being put forward to accept the attribution of information processes to everything that exists. This 'paradigm shift' greatly strengthens the position of computing and associated computational thinking.

In fact, let us try to build ideas in such a way that retains the advantages of both approaches to solving AI problems — cognitive modeling and computational thinking. The necessary generality, as it turns out, is achieved when, as is customary in computing, we consider *information processes* — their emergence, development and distribution, as well as interaction. This would require areas where processes can 'live' — domains or even variable domains. In addition, there are channels through which processes are distributed from region to region.

E. Wigner noted that mathematics is effective in the natural sciences (Wigner, 1960); logic is effective in computer science, and this was defended by many, including J. Halpern, R. Harper and others (Halpern, Joseph Y. and Harper, Robert and Immerman, Neil and Kolaitis, Phokion G. and Vardi, Moshe Y. and Vianu, Victor) (Halpern et al., 2001). Nowadays, in the world of computing, one has to study information processes, using computational thinking. But thanks the efforts of F. William Lawvere and Stephen J. Schanuel we have started to believe that, as it turns out, *conceptual mathematics* (Lawvere & Schanuel, 1997) reveals much of its effectiveness, which is gradually making its way into the knowledge industries.

The philosophical discoveries crystallized in it, which still drive this subject, include the idea that the category of objects of thought is not defined until the category of mappings (maps) that translate these objects into each other and through which they can be compared and distinguished. Thus, in order for mathematics to become applicable and working, going to objects means establishing mappings (Lawvere, 1991).

The crystallized philosophical discoveries which still propel this subject include the idea that a category of objects of thought is not specified until one has specified the category of maps which transform these objects into one another and by means of which they can be compared and distinguished. Thus, for applications of mathematics, to objectify is to mapify.

In fact, the idea that there must be certain domains and certain codomains and that there must be identical mappings is also quite non-trivial. In addition, *processes* must have *states*. It is necessary, more than in any other science, to hold a given object with absolute precision in order to construct, calculate, and draw conclusions; however, we also need to constantly convert it to other objects. The concepts of *existence* (being) and *evolving* (becoming), respectively, will be needed, with existence providing the basis for becoming, as well as a specific mathematical formulation of the principle of 'unity and identity of opposites'. Another idea is the unity and coherence of space, which implies the following condition for the category: 'every object can be included in its associated object'. Considering mappings between objects from a certain 'viewing angle' as a control or deformation process brings certain benefits. Under certain conditions, $A \to T$ can be considered as a state in which T is located (a variable element from T, changing along A), and the composition $A' \xrightarrow{f} A \xrightarrow{h} T$ — as the state h', for $A' \xrightarrow{h'} T$, into which h 'becomes' or 'sets-up' in process f.

The basic idea underlying all the others is that of the category, the 'mathematical universe'. There are many categories, each corresponding to a specific subject area, and there are ways to move from one category to another. The ingredients, as we shall see, are objects, maps (mappings), and map compositions. Although this idea that mathematics includes various categories and their relationships has existed implicitly for centuries, it was not until 1945 that Eilenberg and MacLane gave clear definitions of the basic concepts.

4. Cognitive stage A

Let us start with an informal introduction to views and some examples. The ingredients of the cognitive model will be properties, worlds and correlations that form a referentially clear picture of what is happening.

Let Smith s be unmarried g^-. Consider the marriage of g^+ Smith s, who changes his name r upon marriage, to Jones j, and it is known that in the world A there was already Jones j, which consists married g^+. Building a mathematical model is known as the "Agile Smith problem" (Wolfengagen, 2010).

We have the world A, its inhabitants are Smith s and Jones j, their individuating functions are 'unmarried' g^- and 'married' g^+. To recognize the inhabitants of A as individuals means to represent them as *processes*. Of course, the living Smith and Jones have a lot of aspects by which they can be recognized, but let us focus on the information component, so we consider information processes to be their legitimate representatives.

4.1. Properties

So we are not dealing with individuals themselves, but with processes that represent individuals. It is important to recognize that individuals or their views can be combined into one region, a set which we will call T. This set given in advance, and this is important. It is not necessary that the elements of T be given in constructive sense, since given in advance can only be some property defining the set T. Further, the set T can be separated by imposing restrictions. For example, the properties 'married' g^+ or 'unmarried' g^- can act as such discriminators.

In order to introduce processes, we use the functions

$$\begin{aligned} g^+ &\quad : \quad T \to T^+, \\ g^- &\quad : \quad T \to T^-, \end{aligned} \tag{1}$$

where T^+ is the set of married individuals and T^- is the set of unmarried individuals. Of course, we are talking about the representations of individuals. As it turns out, we can meaningfully compare any two persons existing at different times, and not always two definite persons.

4.2. Worlds

The idea of worlds seems worthy, but we use an extension of this notion, which makes the method much more flexible. This extension will be to use the Carnap–Bar-Hillel principle of indexed expressions. Possible worlds, understood as different collections of individuals with or without additional structure, express only one aspect of this idea. It is important that any system of structures can be indexed by elements of a suitable many, and in different ways. It is clear that the same statement can be true one moment and false the next. In a more general case, indices should not be treated as something as simple as moments in time or possible worlds.

Indices can be called *reference points*, so how to determine the truth value of an expression it is necessary to 'match' it with a point from the set of all indices. As can be seen that this terminology is more neutral and more general than possible worlds. 'Index' also a good term, but it is tied in meaning to the explanatory system that is a characteristic of computational thinking.

5. Reference points

The link system comes naturally. The cognitive characterization of stage A is obtained by choosing the representations of the participating individuals and the connections between them. Take some 'moment' $a \in A$ and some individual $h : A \to T$. The fact that h has the property g^- where $g^- : T \to T^-$, has to be written for $h(a) \in T$ by $(g^- \circ h)(a) \in T^-$. In other words, we get $s = g^- \circ h$. Arguing similarly,

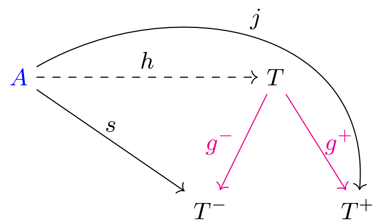

Fig. 1. Generic stage.

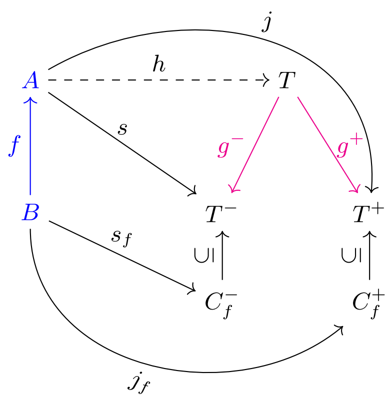

Fig. 2. Evolving the events.

the fact that h has the property g^+ where $g^+ : T \to T+$, write for $h(a) \in T$ by $(g^+ \circ h)(a) \in T^+$. In other words, we get $j = g^+ \circ h$.

This 'sum of knowledge' has an equational characteristic

$$
\begin{array}{rcl}
h & : & A \to T, \\
s & : & A \to T^-, \\
j & : & A \to T^+, \\
g^- & : & T \to T^-, \\
g^+ & : & T \to T^+, \\
s & = & h \circ g^-, \\
j & = & h \circ g^+.
\end{array}
\tag{2}
$$

The resulting commutative diagram is given on Fig. 4 (see Fig. 1).

6. Cognitive stage B

If we are guided by the principles of computational thinking, then we have a 'knowledge stage' A and a 'later' stage B. Let events progress from A to B 'along' f, where $f : B \to A$. At the B stage, the individual $s = g^- \circ h$ transforms to $g^- \circ s_f$ and lives in $C_f^- \subseteq T^-$, while the individual $j = g^+ \circ h$ transforms to $g^+ \circ j_f$ and lives in $C_f^+ \subseteq T^+$. They are treated as f-migrants, as events unfold 'along' f. This is shown in Fig. 2, where the corresponding commutative diagram is shown. But one more event is not taken into account, namely — r-change of property s_f at the same stage B. Then $s_f = g^- \circ h$ becomes $r \circ s_f = r \circ g^- \circ h$, which is accompanied by its r-migration to C_f^+. This is reflected in Fig. 3, which shows the corresponding commutative diagram. In itself, this is interesting in that there was a 'merger' of the two processes $r \circ s_f$ and j_f. For meaningful reasons, we write

$$T^- = T \setminus T^+,$$

and

$$r \circ s_f = j_f = \text{'behave like } j_f\text{'},$$

$$g^- \circ h = s = \text{'behave like } s\text{'},$$

$$g^+ \circ h = j = \text{'behave like } j\text{'}.$$

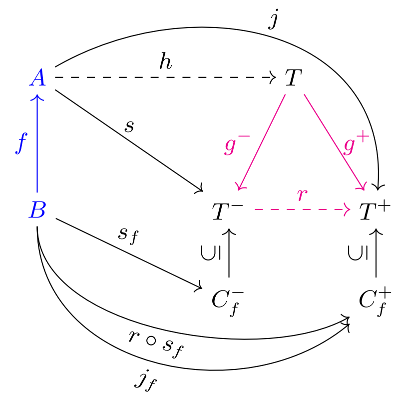

Fig. 3. Change of properties.

Came to the idea that two individuals who, generally speaking, are different, may have the same properties (of a certain kind!) in a given world (relative to given index). Therefore, they are equivalent or randomly incident at this point. Relative to others points of reference, they may not coincide. Allowed branching and merging of individuals — at least insofar as this is allowed by distinctions, associated with certain types of properties. It is possible that confusion about the fusion of individuals stems from the question of allowed properties if the class of properties indefinite, changing, then equivalence individuals may seem paradoxical. Account is for agree to a class of properties given in advance, or by at least agree that this concept is relativized with respect to a fixed class of properties, that we want to explore.

7. Domain map: Cognitive stage A

$H_T(A)$ is inhabited by individuals $h : A \to T$, so we accept the definition

$$H_T(A) = \{h | h : A \to T\}. \tag{3}$$

When the properties $g^+ : T \to T^+$ $g^- : T \to T^-$ arise, then the domains are

$$
\begin{array}{l}
H_{T^-}(A) = \{g^- \circ h | g^- \circ h : A \to T^-\}, \\
H_{T^+}(A) = \{g^+ \circ h | g^+ \circ h : A \to T^+\}.
\end{array}
\tag{4}
$$

In addition, cross-domain links are established

$$
\begin{array}{l}
H_{g^-}(A) : H_T(A) \to H_{T^-}(A) : h \mapsto g^- \circ h, \\
H_{g^+}(A) : H_T(A) \to H_{T^+}(A) : h \mapsto g^+ \circ h.
\end{array}
\tag{5}
$$

This means that on stage A individuals h who lived to $H_T(A)$, either g^--moved to $H_{T^-}(A)$, or to $H_{T^+}(A)$. Note that first the domain $H_T(A)$ appears, which is the *generic* one, and then the domains $H_{T^-}(A)$ and $H_{T^+}(A)$ appear, which are *derived*. A map of variable domains is shown in Fig. 4.

8. Domain map: Cognitive stage B

We keep the domains and channels of the A stage, while supplementing the emerging cognitive model with domains, concepts and channels of the B stage.

8.1. Domains, evolvents and concepts

At the B stage, the $H_T(B)$, $H_{T^-}(B)$, $H_{T^+}(B)$ domains arise:

$$
\begin{array}{rcl}
H_T(B) & = & \{h | h : B \to T\}, \\
H_{T^-}(B) & = & \{h' | h' : B \to T^-\}, \\
H_{T^+}(B) & = & \{h'' | h'' : B \to T^+\}.
\end{array}
\tag{6}
$$

Since stages progress 'along' $f : B \to A$ *from* stage A *to* 'later' stage B (notice the reverse order!), and this progress affects individuals, then

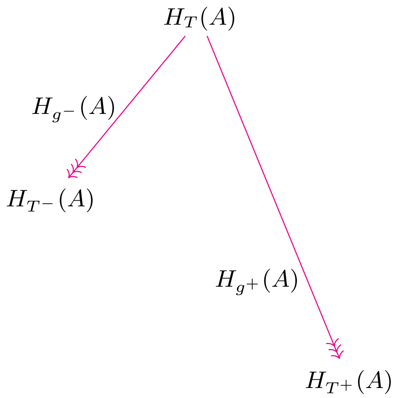

Fig. 4. Agile Smith. Step 1.

sets of individual processes are generated, which are assembled into f-generated domains, so that, in addition to the original domains, on the basis of derived domains, also the concepts $C_f^{1_T} = C_f$, $C_f^{g^-}$, $C_f^{g^+}$, which are subsets of $H_T(B)$, $H_{T^-}(B)$ and $H_{T^+}(B)$ respectively, arise where $C_f \subseteq H_T(B)$, $C_f^{g^-} \subseteq H_{T^-}(B)$, and $C_f^{g^+} \subseteq H_{T^+}(B)$:

$$\begin{array}{lcl} C_f & = & \{h_f | h_f : B \to T\}, \\ C_f^{g^-} & = & \{h' | h' = h\circ f : B \to T^-\}, \\ C_f^{g^+} & = & \{h'' | h'' = h\circ f : B \to T^+\}. \end{array} \tag{7}$$

So way, for domains $H_T(B)$, $H_{T^-}(B)$ and $H_{T^+}(B)$ concepts C_f, $C_f^{g^-}$ and $C_f^{g^+}$ define derived subdomains, respectively.

8.2. g^-- and g^+-channels

The cognitive model would remain incomplete if interdomain process transitions are not taken into account:

$$\begin{array}{lcl} H_{g^-}(A) & : & H_T(A) \to H_{T^-}(A), \\ H_{g^+}(A) & : & H_T(A) \to H_{T^+}(A); \\ H_{g^-}(B) & : & H_T(B) \to H_{T^-}(B), \\ H_{g^+}(B) & : & H_T(B) \to H_{T^+}(B). \end{array} \tag{8}$$

8.3. f-channeling of a process and spawning its clone

Changing a process along an f-channel does not change its properties, but an f-clone of the process appears:

$$\begin{array}{lcl} H_T(f) & : & H_T(A) \to C_f, \\ H_{T^-}(f) & : & H_{T^-}(A) \to C_f^{g^-}, \\ H_{T^+}(f) & : & H_{T^+}(A) \to C_f^{g^+}. \end{array} \tag{9}$$

Thus, for the process h its f-clone h_f is generated, but depending on the context:

$$\begin{array}{lcl} H_T(f) & : & H_T(A) \ni h \mapsto h_f \in C_f, \\ H_{T^-}(f) & : & H_{T^-}(A) \ni g^-\circ h \mapsto g^-\circ h_f \in C_f^{g^-}, \\ H_{T^+}(f) & : & H_{T^+}(A) \ni g^+\circ h \mapsto g^+\circ h_f \in C_f^{g^+}. \end{array} \tag{10}$$

Now let us put together the domains at the stages A and B, the concepts generated by the evolvent $f : B \to A$, and all the discovered channels. This is shown in Fig. 5, which shows — as yet an intermediate cognitive model. It is characterized by a commutative diagram.

8.4. r-channeling of a process

The commutative diagram in Fig. 5 is still not complete, because possible r process transitions are not taken into account, and again this is due to a property change. So at stage B r-channel is added $H_r(1_B)$:

$$\begin{array}{lcl} H_r(1_B) & : & C^{g^-} \to C^{g^+}, \\ H_r(1_B) & : & C^{g^-} \ni g^-\circ h_f \mapsto r\circ g^-\circ h_f \in C^{g^+}. \end{array} \tag{11}$$

9. Closures of channels

We complete the construction of the cognitive model, taking into account the following considerations.

In C_f are located those who moved from A to B. For A-existing ones there are f-indexers, among them there can be g^+- and g^--indexers; they set-up the domains $C_f^{g^-}$ and $C_f^{g^+}$, which are subdomains of $H_{T^-}(B)$ and $H_{T^+}(B)$ respectively.

A channel arises

$$H_{g^-}(f)\circ H_r(1_B) = H_{r\circ g^-}(1_B\circ f) = H_{r\circ g^-}(f), \tag{12}$$

along which the g^--indexers fall into the same domain as the g^+-indexers that went through the 'legal' channel $H_{g^+}(f)$. Thus, the *entanglement* condition becomes

$$H_{g^-}(f)\circ H_r(1_B) = H_{g^+}(f) \tag{13}$$

provided that

$$C_f^{g^+} \cap H_{T^+}(B) \neq \emptyset. \tag{14}$$

This is given in the commutative diagram shown in Fig. 6. Emerging channels from stage A to stage B for Smith

$$H_{g_-}(A)\circ H_{T^-}(f)\circ H_r(1_B) = H^{g^-\circ 1_{T^-}\circ r}(1_B\circ f\circ 1_A) \tag{15}$$

and for Jones

$$H_{g^+\circ 1_{T^+}}(f\circ 1_A) = H_{g_+}(A)\circ H_{T^+}(f)\circ \subseteq \tag{16}$$

characterize the corresponding processes.

10. Discussion in brief

This paper attempts to build a cognitive computational model and gives a framework based on F. Lawvere's conceptual mathematics. This is possibly a distinction of the present work among a variety of AI cognitive frameworks. The best way would be to summarize the results we achieved with additional comments.

Why semantic information can be destroyed? As we noted the individuals in the theory are not the individual constants but are assumed as processes. To illustrate the idea we use the 'Agile Smith task'. A person Smith when identified is perceived as a process because of possible variability of his properties. There is some difficulty to describe him in, say, database system without violation of integrity constraints. There is a strict chance to retrieve some other person, say, Jones with the same properties as 'later' described Smith ('later' here means 'at other stage of knowledge'). A sample diagram reflects possible *entanglement* of the associated information processes.

This effect of entanglement was established and partially studied by the authors. The conclusion was that we are not able just avoid it using the elementary logic means, roughly speaking, because of the variability of individual constants. At least, we need a non-trivial referential machinery with possible worlds or alike. But we are able to construe the representation. The model represented is based on Lawvere's construction of a variable domain. The details can be successfully found in his papers.

The most straightforward method to model entanglement of information processes is to use the diagrams of 'conceptual dependencies' as the relationships between variable domains. The indices in their names have the rich semantic sense and meaning. This ability is not yet widely used.

This is a commonly used practice of using the computational thinking. We use P. Denning's treatment of computational thinking, and this way is followed now by ACM in a rather wide area. Our contribution is in some particular selection of 'pattern recognition' — we operate with the representations of information processes as we assume symbolic representations of individuals as varying over stages of knowledge.

The conceptual graphs over variable domains have the advantage of strict mathematical representation. Fortunately, this is a kind of functor

$$\begin{tikzcd}
& H_T(A) \arrow[rr, two heads, "H_T(f)"] \arrow[dl, two heads, "H_{g^-}(A)"'] \arrow[ddr, two heads, "H_{g^+}(A)"'] & & C_f \arrow[rr, two heads, "\subseteq"] \arrow[dl, two heads, "H_{g^-}(1_B)"'] \arrow[ddr, two heads, "H_{g^+}(1_B)"'] & & H_T(B) \arrow[dl, two heads, "H_{g^-}(B)"'] \arrow[ddr, two heads, "H_{g^+}(B)"'] & \\
H_{T^-}(A) \arrow[rr, two heads, "H_{T^-}(f)"] & & C_f^{g^-} \arrow[rr, two heads, "\subseteq"] & & H_{T^-}(B) & & \\
& & H_{T^+}(A) \arrow[rr, two heads, "H_{T^+}(f)"] & & C_f^{g^+} \arrow[rr, two heads, "\subseteq"] & & H_{T^+}(B)
\end{tikzcd}$$

Fig. 5. Agile Smith. Step 2.

$$\begin{tikzcd}
& H_T(A) \arrow[rr, two heads, "H_T(f)"] \arrow[dl, two heads, "H_{g^-}(A)"'] \arrow[ddr, two heads, "H_{g^+}(A)"'] \arrow[dr, Rightarrow, dashed, "H_{g^-}(f)"] \arrow[ddrrr, Rightarrow, dashed, bend right, "H_{g^-}(f) \circ H_r(1_B)"'] \arrow[ddrrrrr, Rightarrow, dashed, bend right=60, "H_{g^+}(f)"'] & & C_f \arrow[rr, two heads, "\subseteq"] \arrow[dl, two heads, "H_{g^-}(1_B)"'] \arrow[ddr, two heads, "H_{g^+}(1_B)"'] & & H_T(B) \arrow[dl, two heads, "H_{g^-}(B)"'] \arrow[ddr, two heads, "H_{g^+}(B)"'] & \\
H_{T^-}(A) \arrow[rr, two heads, "H_{T^-}(f)"] & & C_f^{g^-} \arrow[rr, two heads, "\subseteq"] \arrow[ddrr, dashed, two heads, "H_r(1_B)"'] & & H_{T^-}(B) \arrow[ddrr, dashed, two heads, "H_r(B)"'] & & \\
& & H_{T^+}(A) \arrow[rr, two heads, "H_{T^+}(f)"] & & C_f^{g^+} \arrow[rr, two heads, "\subseteq"] & & H_{T^+}(B)
\end{tikzcd}$$

Fig. 6. Agile Smith. Step 3.

category equipped with natural transformations. This category is used as a valid universe of discourse. Some authors have started studying of this approach but we mean to attract more attention of the Cognitive Systems Research community.

What investigate next? For instance we see a road to construe the non-standard conceptual modeling to operate information processes channeling. Thus, we need, e.g., to construe the property inheritance over variable domains. The legacy of variable properties looks as a good case.

Other direction is to learn more lessons on cognition of displaced concepts. This would be a step towards a kind of cognitive abstract machine if any.

11. Conclusions

The constructions presented in this paper are rather cumbersome, so it would be appropriate to draw conclusions related to what has been achieved, as well as to indicate the results obtained.

1. An effort has been made to characterize computational thinking. Notable points are ordered and emphasized in a slightly different way so that the connection with cognitive modeling becomes more tangible. This became possible as a result of following the basic premises of conceptual mathematics.

2. On this path, individuals are perceived as processes, and the main difficulty lies in maintaining the intuitive justification of such a representation. Some deep essence of what is happening is preserved, and within the framework of computational thinking, a focus on operating information processes is ensured. The representations of individuals are transformed into processes, then it remains to learn what happens to them at different stages.

3. How do you need to reason in order to achieve, in order to perceive the variability of processes and what happens to them — there is no strict rule. What matters is the referentiality of processes and their referential clarity. According to the principles of conceptual mathematics, the behavior of processes with respect to their composition is studied. They also act when constructing the semantics of programming languages, computing is implemented through its practices — programming, which is modeled by compositions on the hidden state of the computation.

4. The transition to the domains in which individuals and the processes that represent them live sharply raises the main question for the resulting cognitive model — how the interdomain exchange of processes occurs and how they are transformed in this case. The idea of a channeled going of processes arises, and cognitive activity is directed to the study of emerging flows.

5. In short, the conceptual transform of computational thinking generates a completely meaningful cognitive model of what is happening,

not losing connection with what is happening with the processes, in this sense it is constructive.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

References

Bangu, S. (2016). On 'the unreasonable effectiveness of mathematics in the natural sciences'. In E. Ippoliti, F. Sterpetti, & T. Nickles (Eds.), *Models and inferences in science* (pp. 11–29). Cham: Springer International Publishing, http://dx.doi.org/10.1007/978-3-319-28163-6_2.

Brachman, R. J. (1977). What's in a concept: Structural foundations for semantic networks. *International Journal of Man-Machine Studies*, *9*(2), 127–152. http://dx.doi.org/10.1016/S0020-7373(77)80017-5.

Brachman, R. (2006). (AA) AI more than the sum of its parts. *AI Magazine*, *27*, 19–34. http://dx.doi.org/10.1609/aimag.v27i4.1907.

Brüssow, S., Ragni, M., Frorath, M., Konieczny, L., & Fangmeier, T. (2013). Premise annotation in mental model construction: An ACT-R approach to processing indeterminacy in spatial relational reasoning. *Cognitive Systems Research*, *24*, 52–61. http://dx.doi.org/10.1016/j.cogsys.2012.12.011, URL https://www.sciencedirect.com/science/article/pii/S1389041712000642. Cognitive Systems Research:Special Issue on ICCM2012.

Cassimatis, N. L. (2012). Artificial intelligence and cognitive modeling have the same problem. In P. Wang, & B. Goertzel (Eds.), *Theoretical foundations of artificial general intelligence* (pp. 11–24). Paris: Atlantis Press, http://dx.doi.org/10.2991/978-94-91216-62-6_2.

Denning, P. J., & Tedre, M. (2019). *Computational thinking*. MIT PressThe MIT Press, URL http://gen.lib.rus.ec/book/index.php?md5=A8A4B175F4F4BB07E40AF0EED5A804DE.

Denning, P. J., & Tedre, M. (2021). Computational thinking for professionals. *Communications of the ACM*, *64*(12), 30–33. http://dx.doi.org/10.1145/3491268.

Feigenbaum, E., & McCorduck, P. (1983). *The fifth generation: artificial intelligence and Japan's computer challenge to the world*. USA: Addison-Wesley Longman Publishing Co., Inc..

Goto, E., Soma, T., Inada, N., Ida, T., Idesawa, M., Hiraki, K., et al. (1982). Design of a lisp machine - FLATS. In *Proceedings of the 1982 ACM symposium on LISP and functional programming* (pp. 208–215). New York, NY, USA: Association for Computing Machinery, http://dx.doi.org/10.1145/800068.802152.

Gyöngyi, Z., Garcia-Molina, H., & Pedersen, J. (2004). Combating web spam with TrustRank. In M. A. Nascimento, M. T. Özsu, D. Kossmann, R. J. Miller, J. A. Blakeley, & B. Schiefer (Eds.), *Proceedings 2004 VLDB conference* (pp. 576–587). St Louis: Morgan Kaufmann, http://dx.doi.org/10.1016/B978-012088469-8.50052-8, URL https://www.sciencedirect.com/science/article/pii/B9780120884698500528.

Halpern, J. Y., Harper, R., Immerman, N., Kolaitis, P. G., Vardi, M. Y., & Vianu, V. (2001). On the unusual effectiveness of logic in computer science. *Bulletin of Symbolic Logic*, *7*(2), 213–236. http://dx.doi.org/10.2307/2687775.

Hölken, A., Kugele, S., Newen, A., & Franklin, S. (2023). Modeling interactions between the embodied and the narrative self: Dynamics of the self-pattern within LIDA. *Cognitive Systems Research*, *81*, 25–36. http://dx.doi.org/10.1016/j.cogsys.2023.03.002, URL https://www.sciencedirect.com/science/article/pii/S1389041723000335.

Hughes, C., & Hughes, T. (2019). The laws of thought and thinking machines. *AI Matters*, *5*(1), 20–24. http://dx.doi.org/10.1145/3320254.3320263.

Ismailova, L., Wolfengagen, V., & Kosikov, S. (2022a). Applicative approach to construe a computational model of concepts and individuals. *Procedia Computer Science*, *213*, 463–470. http://dx.doi.org/10.1016/j.procs.2022.11.092, URL https://www.sciencedirect.com/science/article/pii/S1877050922017835. 2022 Annual International Conference on Brain-Inspired Cognitive Architectures for Artificial Intelligence: The 13th Annual Meeting of the BICA Society.

Ismailova, L., Wolfengagen, V., & Kosikov, S. (2022b). Conceptual hanger for displaced concepts: a framework for information processes variability. *Procedia Computer Science*, *213*, 588–595. http://dx.doi.org/10.1016/j.procs.2022.11.109, URL https://www.sciencedirect.com/science/article/pii/S1877050922018087. 2022 Annual International Conference on Brain-Inspired Cognitive Architectures for Artificial Intelligence: The 13th Annual Meeting of the BICA Society.

Ismailova, L., Wolfengagen, V., & Kosikov, S. (2022c). Elements of semantic analysis based on lambda-calculus. *Procedia Computer Science*, *213*, 471–476. http://dx.doi.org/10.1016/j.procs.2022.11.093, URL https://www.sciencedirect.com/science/article/pii/S1877050922017847. 2022 Annual International Conference on Brain-Inspired Cognitive Architectures for Artificial Intelligence: The 13th Annual Meeting of the BICA Society.

Ismailova, L., Wolfengagen, V., & Kosikov, S. (2022d). Lambda-calculus, combinators and applicative computational technologies. *Cognitive Systems Research*, *76*, 93–100. http://dx.doi.org/10.1016/j.cogsys.2022.10.002, URL https://www.sciencedirect.com/science/article/pii/S1389041722000468.

Ismailova, L., Wolfengagen, V., & Kosikov, S. (2022e). A prototype system for supporting a network of information graphs with the ability to assess the nature of the subject's knowledge. *Procedia Computer Science*, *213*, 16–20. http://dx.doi.org/10.1016/j.procs.2022.11.033, URL https://www.sciencedirect.com/science/article/pii/S1877050922017240. 2022 Annual International Conference on Brain-Inspired Cognitive Architectures for Artificial Intelligence: The 13th Annual Meeting of the BICA Society.

Ismailova, L., Wolfengagen, V., Kosikov, S., & Andronov, S. I. (2023). The applicative approach to the synthesis of a data structure with the given combinatory characteristic. *Cognitive Systems Research*, *77*, 88–93. http://dx.doi.org/10.1016/j.cogsys.2022.10.010, URL https://www.sciencedirect.com/science/article/pii/S1389041722000602.

Johannes José Fijen, L., Joaquín López González, J., & Treur, J. (2021). An adaptive temporal-causal network model to analyse extinction of communication over time. *Cognitive Systems Research*, *68*, 73–83. http://dx.doi.org/10.1016/j.cogsys.2020.08.013, URL https://www.sciencedirect.com/science/article/pii/S1389041721000231.

Kangasrääsiö, A., Athukorala, K., Howes, A., Corander, J., Kaski, S., & Oulasvirta, A. (2017). Inferring cognitive models from data using approximate Bayesian computation. In *Proceedings of the 2017 CHI conference on human factors in computing systems* (pp. 1295–1306). New York, NY, USA: Association for Computing Machinery, http://dx.doi.org/10.1145/3025453.3025576.

Kangasrääsiö, A., Jokinen, J. P. P., Oulasvirta, A., Howes, A., & Kaski, S. (2019). Parameter inference for computational cognitive models with approximate Bayesian computation. *Cognitive Science*, *43*(6), Article e12738. http://dx.doi.org/10.1111/cogs.12738.

Kugele, S., & Franklin, S. (2021). Learning in LIDA. *Cognitive Systems Research*, *66*, 176–200. http://dx.doi.org/10.1016/j.cogsys.2020.11.001, URL https://www.sciencedirect.com/science/article/pii/S1389041720300826.

Lawvere, F. W. (1991). Some thoughts on the future of category theory. In A. Carboni, M. C. Pedicchio, & G. Rosolini (Eds.), *Category theory* (pp. 1–13). Berlin, Heidelberg: Springer Berlin Heidelberg.

Lawvere, F. W., & Schanuel, S. J. (1997). *Conceptual mathematics: a first introduction to categories*. Cambridge: Cambridge University Press.

Marquis, J.-P. (2015). Category theory. In E. N. Zalta (Ed.), *The stanford encyclopedia of philosophy* (Winter 2015 ed.).

McCarthy, J. (1978). History of LISP. *SIGPLAN Notices*, *13*(8), 217–223. http://dx.doi.org/10.1145/960118.808387.

McCarthy, J., Minsky, M. L., Rochester, N., & Shannon, C. E. (1955). A proposal for the dartmouth summer research project on artificial intelligence. URL http://www-formal.stanford.edu/jmc/history/dartmouth/dartmouth.html.

Richard Moore, L., & Gunzelmann, G. (2014). An interpolation approach for fitting computationally intensive models. *Cognitive Systems Research*, *29–30*, 53–65. http://dx.doi.org/10.1016/j.cogsys.2013.09.001, URL https://www.sciencedirect.com/science/article/pii/S1389041713000508.

Saptawijaya, A., & Pereira, L. M. (2015). The potential of logic programming as a computational tool to model morality. In R. Trappl (Ed.), *A construction manual for robots' ethical systems: requirements, methods, implementations* (pp. 169–210). Cham: Springer International Publishing, http://dx.doi.org/10.1007/978-3-319-21548-8_10.

Schubert, H. (1972). *Categories*. Berlin, Heidelberg: Springer Berlin Heidelberg.

Slieptsov, I. O., Wolfengagen, V. E., & Kosikov, S. V. (2022). Constructing generator of words of context-sensitive language on example of typed λ-calculus. *Procedia Computer Science*, *213*, 556–562. http://dx.doi.org/10.1016/j.procs.2022.11.104, URL https://www.sciencedirect.com/science/article/pii/S1877050922017951. 2022 Annual International Conference on Brain-Inspired Cognitive Architectures for Artificial Intelligence: The 13th Annual Meeting of the BICA Society.

Spivak, D. I. (2014). *Category theory for the sciences*. The MIT Press.

Stewart, T. C., & West, R. L. (2007). Deconstructing and reconstructing ACT-R: Exploring the architectural space. *Cognitive Systems Research*, *8*(3), 227–236. http://dx.doi.org/10.1016/j.cogsys.2007.06.006, URL https://www.sciencedirect.com/science/article/pii/S1389041707000253. Cognitive Modeling.

Turing, A. M. (1950). Computing machinery and intelligence. *Mind*, *LIX*(236), 433–460. http://dx.doi.org/10.1093/mind/LIX.236.433.

Vallverd, J., & Vallverd, J. (2010). *Thinking machines and the philosophy of computer science: concepts and principles*. Hershey, PA: Information Science Reference - Imprint of: IGI Publishing.

Weinreb, D., & Moon, D. (1981). The lisp machine manual. *SIGART Bulletin*, (78), 10. http://dx.doi.org/10.1145/1056737.1056738.

Wigner, E. (1960). The unreasonable effectiveness of mathematics in the natural sciences. *Communications on Pure and Applied Mathematics*, *13*(1), 1–14. http://dx.doi.org/10.1002/cpa.3160130102.

Wolfengagen, V. (2010). Semantic modeling: Computational models of the concepts. In *2010 international conference on computational intelligence and security* (pp. 42–46). http://dx.doi.org/10.1109/CIS.2010.16.

Wolfengagen, V., Ismailova, L., & Kosikov, S. (2022). Cognitive technology to capture deep computational concepts with combinators. *Cognitive Systems Research*, *71*, 9–23. http://dx.doi.org/10.1016/j.cogsys.2021.10.001, URL https://www.sciencedirect.com/science/article/pii/S1389041721000747.

Wolfengagen, V., Ismailova, L., Kosikov, S., & Babushkin, D. (2021). Modeling spread, interlace and interchange of information processes with variable domains. *Cognitive Systems Research*, *66*, 21–29. http://dx.doi.org/10.1016/j.cogsys.2020.10.016, URL https://www.sciencedirect.com/science/article/pii/S1389041720300905.

Woods, W. A. (1975). What's in a kink: Foundations for semantic networks. In D. G. Bobrow, & A. Collins (Eds.), *Representation and understanding* (pp. 35–82). San Diego: Morgan Kaufmann, http://dx.doi.org/10.1016/B978-0-12-108550-6.50007-0, URL http://www.sciencedirect.com/science/article/pii/B9780121085506500070.

Yan, B., Yang, C., Shi, C., Fang, Y., Li, Q., Ye, Y., et al. (2023). Graph mining for cybersecurity: A survey.