Published in the Slovak Republic
European Journal of Technology and Design
Has been issued since 2013.
E-ISSN: 2310-3450
2019, 7(1): 3-7

DOI: 10.13187/ejtd.2019.1.3
www.ejournal4.com

Articles

Logical Sequences

Pavel Yu. Elsukov [a, *]

[a] Federal State Institution of Science Institute of Energy Systems. LA Melentyeva, Siberian Branch of the Russian Academy of Sciences (ESI SB RAS), Irkutsk, Russian Federation

Abstract

The article analyzes logical sequences used in logic and computing. The main objective of the research is to show the relationship between logic, computing and functional expressions. Existence of such connection allows to make a comprehensive analysis of formulas and analytic descriptions. Comprehensive analysis simplifies consistency check of formulas. Existence of relationship between three types of descriptions allows to analyze semi-structured information. The article reveals the contents of a logical sequence used for research. In this article, logical sequence is a general concept related to logic and mathematics.

Keywords: artificial intelligence, logic, logical sequences, linear logical sequences, parallel logical sequences.

1. Introduction

Construction of logical sequences or logical chains (Raev, Tsvetkov, 2018) is commonly used in compiling algorithms and in scientific projects. Construction of logical sequences is used in preparation of doctoral dissertations. When performing research projects, the aim is to conduct sound research and to achieve outcome studies, which refute or prove initial theses. This task is accomplished by constructing and applying logical sequences. When preparing doctoral dissertations, the aim is to conduct sound research and to rationalize new scientific solutions to tasks. Validity of conclusions also requires the use of logical sequences. The essence of logical sequences construction is summarized in this article.

2. Materials and methods

Numerous publications and scientific reports on logic and scientific analysis are used as materials. Mathematical logic and system analysis are used as methods.

3. Results

General Formulation of Research

Logical sequence is a general concept that incorporates a sequence of computations, a sequence of functional transformations and logical chains. Logical consequence (Tarski, 1936) is the most simple logical chain. For this reason, logical consequence is the basis of a logical

* Corresponding author
E-mail addresses: elsukov_p@bk.ru (P.Yu. Elsukov)

sequence. There are simple and complex logical consequences.

Simple logical consequence is a relation between one premise A and one conclusion B. Implication (Ebbinghaus et al., 2013) describes a simple logical consequence.

$$A \rightarrow B. \quad (1)$$

or

$$A_1 \rightarrow A_2 \quad (2)$$

Expression (2) describes information situation (Tsvetkov, 2012), in which state A_1 implies state A_2. Complex logical consequence describes set of transitions of type (2). The simplest logical sequence is expressed with a complex logical consequence. Logical sequence can describe not only transitions, but states as well. In actual practice the system or algorithm can be consistently in different states. Such situation or logical sequence is shown in Figure 1.

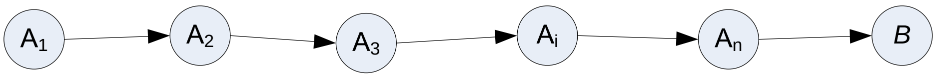

Fig. 1. Logical Consequence

Figure 1 shows a complex logical consequence or a simple logical sequence. Complex linear logical consequence (logical sequence) is a set of relations between a chain of premises A_i and one conclusion B. Figure 1 describes expression (3).

$$A_1 \rightarrow A_2 \rightarrow A_3 \rightarrow A_i \rightarrow A_n \rightarrow B. \quad (3)$$

Expression (3) describes a logical sequence, or logical chain, or information construction (Tsvetkov, 2014). Expression (1) describes a link of logical consequence or logical sequence.

Practical formation of logical sequence in research includes several steps. The first step is to formulate the objectives of study "B". The second step is to define key points of the study. Key points of the study form key states "A_i", which are included in expression (3). These key states are shown in Figure 1 as vertices.

The third stage is the most important one. It involves finding relations and state transitions in expression (3). Here we should emphasize the difference between logical relation and functional transformations. If expression (3) is considered as a logical construction, all transitions between "A" states therein are the same and correspond to relation of implication. If expression (3) is considered as functional consequence, all transitions between "A" states can be different. Each transition can correspond to a specific function. Expression (3) can be written using universal quantifier

$$\forall A_i\,(A_{i-1} \rightarrow A_i) \quad (4).$$

Diagram in Figure 1 has a dualinterpretation. One interpretation exists in the area of mathematical logic, the other lies in the area of functional analysis. In the area of logic diagram on Fig. 1 and expression (3) describe the classical Tarski logical consequence (Tarski, 1936). Semantic notion of logical consequence was introduced by Tarski in 1936.Formal definition of the logical consequence has the following form:

$$(A1, \ldots, An) \models B \quad (5)$$

Semantic definition of logical consequence (5) reads as follows: statement B logically follows from premises (A1, ..., An), if it is impossible that statements A1,..., An are true, and statement B is false (i.e. if B is true in any model where A1,..., An are true). The distinctive feature of a logical consequence lies in the fact that it leads from true statements only to true statements.

In the area of computational analysis this means correctness of computations in transition from one function of chain (3) to another function.

$A_i$can be seen as logical formulas in expression (5). In such case there are three conditions for them. 1. There is a set Γ of derivable formulas. 2. There is a subset Δ of derivable formulas. 3. There is a set of formulas Fm, which can be either derivable or non-derivable. These conditions are expressed formally as follows:

$$\Gamma, \Delta \subseteq Fm\,;\, A, B \in Fm \quad (6)$$

If there are initial conditions (6), then there are three properties (7-9) for logical consequence (5).

$$A \in \Gamma \Rightarrow \Gamma \models A \text{ (reflexivity)}, \quad (7)$$

$$\Gamma \models A \text{ and } \Gamma \subseteq \Delta \Rightarrow \Delta \models A, \text{ (monotonicity), (8)}$$
$$\Gamma \models A; \Gamma, A \models B, \Rightarrow \Gamma \models B \text{ (transitivity or separation). (9)}$$

Logical consequence allows to introduce a definition of paraconsistent logic.

Assume that $\models$ is a relation of a logical consequence. Consequence is called explosive, if it meets the condition that for any formulas A and B, Bfollows from $A$and *not-A*.

$$\{A, \neg A\} \models B. \text{ (10)}$$

Classical logic, intuitionistic logic, multi-valued Łukasiewicz logic and most other logics are explosive. Logic is called paraconsistent (da Costa et al., 1991), if and only if its relation of logical consequence *is not* explosive.

Logical sequence shown in Figure 1 is linear and sequential. There are parallel logical sequences. Parallel logical sequence consisting of two chains is shown in Figure 2.

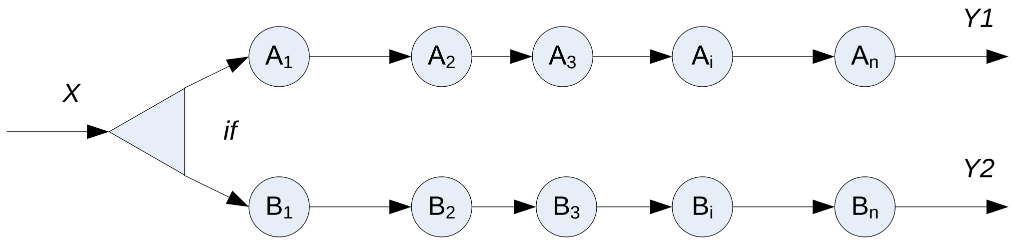

Fig. 2. Parallel logical sequence

There can be any number of parallel chains. Such sequence (Figure 2) is called *acyclic* (Skulrattanakulchai, 2004; Gebremedhin et al., 2007), since there are no cycles in this scheme.

$$X \rightarrow [(\forall A_i (A_{i-1} \rightarrow A_i) \rightarrow Y1) \oplus (\forall B_i (B_{i-1} \rightarrow B_i) \rightarrow Y2)] \text{ (11)}$$

In expression (11) X is an array of input data; $Y1$, $Y2$ are arrays of output data. There can be more than two arrays of output data.

There are no such strict requirements for parallel logical sequences as compared to functional sequences. If we consider the diagram in Figure 2 as an example of parallel computations, then additional conditions emerge for it in computing area: synchronization, data race, deadlock (Boyapat et al., 2002). There are no such conditions in the area of logic (Figure 3).

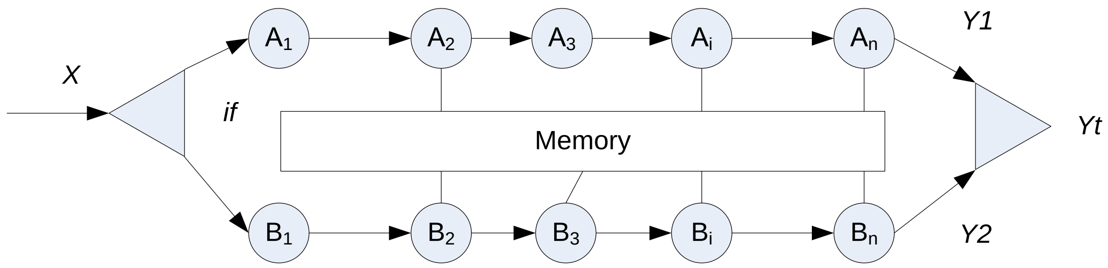

Fig. 3. Logical sequence in parallel information processing

Nodes *A, B* in Figure 3 can be seen as computation modules. Each branch in Figure 3 characterizes each individual processor. If multiple modules of the same program are run on different processors, the need for synchronization of operations emerges. For example, $Y1$, $Y2$ must be obtained simultaneously for transformation into a common array Yt.

Therefore, operation of chain A must be synchronized with operation of chain B. Parallel computing requires careful synchronization of program modules executed by different chains of computations.

Initial branch of input data X is called Fork, while aggregation of data is defined as join. There must be Fork/join (Lea, 2000; Lebrecht, Knottenbelt, 2007) sequence for parallel computations. There is no such sequence in logical consequences. This can be explained by impossibility of solving tasks, where uncertainty arises, by means of binary logic. These situations require the use of ternary logic or additional logical sequences. Fork/join sequence eliminates incompleteness of binary logic.

"Data race" (Narayanasamy et al., 2007) problem emerges for multiprocessor computers with shared memory. If processors operate in parallel, they can access the same data simultaneously. These data are stored in shared memory (Figure 3) for both reading and recording. Processors operating in parallel can simultaneously access the same data stored in shared memory. Simultaneous reading does not cause any problems, while recording two different values into the same memory cell is impossible. Recording must be executed on one-by-one basis. Attempt of simultaneous recording of different values into one memory cell causes data race. Computer memory in Figure 3 creates a branch from logical chains A, B. These branches are directed from each vertex perpendicular to the direction of common logical consequence. Data race problem is one of the most serious problems for parallel programming, because in case of incorrectly organized resource blocking program can be completed with incorrect results.

Deadlock situation (Piroddi et al., 2008). Locking and barring of access to resource for competitors are required in order to cope with data race problem. Competitors mean parallel computing processes. They must interrupt their operation and line up, waiting for release of the resource. Locking is a useful mechanism, which can not be ignored. But locking is also a very dangerous mechanism, which can result in suspension and termination of the program. This situation is called deadlock.

All three problems are not described by binary logic. Therefore, they require additional information structures and expansion of binary logic to ternary logic.

Initial information set X is transformed into a system of interconnected facts, patterns, rules, inferences Y on the basis of logical sequence.

A qualitative difference between input set X and output set Y should be noted. Input set X can be unstructured and unsystematized. Output set Y is structured and systematized. Application of logical sequence structures and systematizes input information.

Eventually, logical sequence creates a knowledge system.

$$Y \to Kn\ (C, E)$$

This knowledge system includes connections (C) and elements (E). But this system has an area of trueness, which is also determined in the course of research.

Input set is only partially included in the knowledge system. A part of input set falls into "non system". In scientific research, output information is divided into reliable information and information uncertainty. "Non system" is divided into "antagonism" and uncertainty. Antagonism designates the part of information, which refutes research task or is contrary to it. Uncertainty requires further analysis. The area of uncertainty is the source of solutions to new tasks and scientific novelty. As a result, initially formed knowledge system expands with new knowledge resulting from solving new scientific tasks.

Basic logical chain or logical sequence is called a forming chain or sequence. It serves as the ground for evidence base. In addition, there are indirect logical chains with support functions.

For example, in scientific research indirect logical sequence includes a set of figures and diagrams. A set of figures and diagrams is an additional logical sequence, which supports and clarifies the logic of presentation and the logic of evidence.

Indirect logical chain is linked to the construction of semantic space of research area or definitions system. Semantic space generally means a set of organized indicators, which describe a certain content area (Raev, Tsvetkov, 2018). Semantic space of a scientific research means combination of keywords, categories and relations between them, which describe content of the research area.

There is a verbal logical sequence in addition to formal logical sequences. This sequence is called discursive logical sequence.

Discursive logical sequence is conditional upon loss of relevance of the research topic during the period of study. New ideas may appear refuting or changing the author's hypothesis until research is complete. Errors in presentation may occur. This conditions discourse as situational evidence and requires introduction of discursive logical sequence as evidence of hypothesis in the light of new facts.

4. Conclusion

Relevance of scientific research and reliability of the results is related to sets of logical sequences. Logical sequences connect logic, mathematics and computing. Logical sequences have different qualities and perform different functions. Some sequences address structuring and systematization problems. Other logical sequence create substantiation of hypothesis. Still other logical sequences serve as support for the hypothesis. In general, the method of logical sequences can be named as a means of mandatory creation of substantiated scientific research. Set of logical sequences is used for conduct of a comprehensive analysis. The method of logical sequences is the developing area requiring further research and development.

References

Boyapat et al., 2002 – *Boyapati, C., Lee, R., Rinard, M.* (2002, November). Ownership types for safe programming: Preventing data races and deadlocks. *ACM Sigplan Notices*, Vol. 37, No. 11, pp. 211-230.

da Costa, et al., 1991 – *da Costa, N.C., Subrahmanian, V.A., Vago, C.* (1991). The Paraconsistent Logics PJ. *Mathematical Logic Quarterly*, 37(9-12), 139-148.

Ebbinghaus et al., 2013 – *Ebbinghaus, H.D., Flum, J., Thomas, W.* (2013). Mathematical logic. Springer Science & Business Media.

Gebremedhin et al., 2007 – *Gebremedhin, A.H., Tarafdar, A., Manne, F., Pothen, A.* (2007). New acyclic and star coloring algorithms with application to computing Hessians. *SIAM Journal on Scientific Computing*, 29(3): 1042-1072.

Lea, 2000 – *Lea, D.* (2000, June). A Java fork/join framework. *Proceedings of the ACM 2000 conference on Java Grande*, pp. 36-43.

Lebrecht, Knottenbelt, 2007 – *Lebrecht, A., Knottenbelt, W.* (2007). Response time approximations in fork-join queues.

Narayanasamy et al., 2007 – *Narayanasamy, S., Wang, Z., Tigani, J., Edwards, A., Calder, B.* (2007, June). Automatically classifying benign and harmful data races using replay analysis. *ACM SIGPLAN Notices*, Vol. 42, No. 6, pp. 22-31.

Piroddi et al., 2008 – *Piroddi, L., Cordone, R., Fumagalli, I.* (2008). Selective siphon control for deadlock prevention in Petri nets. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 38(6): 1337-1348.

Raev, Tsvetkov, 2018 – *Raev V.K., Tsvetkov V.Ya.* (2018). Logical chains. R*emote and virtual learning,* № 1 (120), pp. 14-21. [in Russian]

Skulrattanakulchai, 2004 - Skulrattanakulchai, S. (2004).Acyclic colorings of subcubic graphs. *Information processing letters*, *92*(4), 161-167.

Tarski, 1936 – *Tarski, A.* (1936). On the concept of logical consequence. *Logic, semantics, metamathematics*, 2: 1-11.

Tsvetkov, 2012 – *Tsvetkov, V.Ya.* (2012). Information Situation and Information Position as a Management Tool. *European researcher. Series A*, (12-1): 2166-2170.

Tsvetkov, 2014 – *Tsvetkov, V.Y.* (2014). Information Constructions. *European Journal of Technology and Design*, (3): 147-152.