UDC 0.04+0.07

P.P. TROKHYMCHUK, O.M. VILIHURSKYI, O.M. ZAMURUIEVA
Lesya Ukrainka Volyn National University

MAIN PROBLEMS OF EVOLUTION THE CYBERNETICS AND COMPUTER SCIENCE

The problems of evolution the cybernetics and computer science are analysing. Short historical analysis of this problem is representiung. It is including Greek abacus, the Peruvian system of nodal counting Kipu. The role of Blaise Pascal and Wilhelm Leibniz in establishing the foundations of computer science is noted. The next stage in the development of computer science was the research of Charles Babbage and Lady Ada Lovelace. It was Ada Lovelace, who initiated the programming procedure. The concept of cybernetics as the management of ships originated in Greece. In the 19th century, it was formulated as a science of management by F.-M. Ampere and B. Trentowski. It was completed by N. Wiener, according to whom cybernetics is the science of control in the living and non-living world. Later, cybernetics became the basis of computing. In its bowels, the theory of automatic regulation was expanded and the foundations of modern information theory were formulated. As F. George showed, cybernetics is a synthetic science that includes a number of sciences that are needed to solve the relevant problem. Research has been conducted on the development of the hardware base of modern cybernetics and computer science: from pebbles, nodules and bones to modern optoelectronic systems. Modern computer science has a somewhat broader meaning as defined by N. Wiener. The main task of modern computer science is the formalization of the thesis of the Canadian philosopher L. Hall "Everything that comes from the head is intelligent". In this case, along with the elementary base, programming received significant development. Along with narrow-profile programming languages (Fortran, Pascal), the system programming languages C and cross-hierarchical programming (Python have been created. The structure of computer science has also changed significantly. The further development of computer systems is obviously related to the reduction of time and simplification of the procedure for obtaining the necessary information. Possible ways of implementing this are also discussed.

Key words: cybernetics, computer science, evolution, hardware, software, polymetrical analysis, Python.

П.П. ТРОХИМЧУК, О.М. ВІЛІГУРСЬКИЙ, О.М. ЗАМУРУЄВА
Волинський національний університет імені Лесі Українки

ОСНОВНІ ПРОБЛЕМИ ЕВОЛЮЦІЇ КІБЕРНЕТИКИ ТА КОМП'ЮТЕРНИХ НАУК

У статті проаналізовано проблеми еволюції кібернетики й інформатики. Подано короткий історичний аналіз цієї проблеми. Сюди входить абак, перуанська вузликова система рахунку Кіпу. Відзначається роль Блеза Паскаля й Вільгельма Ляйбніца у створенні основ інформатики. Наступним етапом у розвитку інформатики стали дослідження Чарльза Беббіджа й леді Ади Лавлейс. Саме Ада Лавлейс ініціювала процедуру створення програмування. Поняття кібернетики як управління кораблями виникло в Греції. У 19 столітті воно сформульовано як наука про управління Ф.-М. Ампером і Б. Трентовським. Його завершив Н. Вінер, згідно з яким кібернетика – це наука про управління живим і неживим світом. Пізніше кібернетика стала основою обчислювальної техніки. У її надрах розширено теорію автоматичного регулювання та сформульовано основи сучасної теорії інформації. Як показав Ф. Джордж, кібернетика є синтетичною наукою, яка включає низку наук, необхідних для розв'язання відповідної проблеми. Проведено дослідження щодо розвитку апаратної бази сучасної кібернетики й інформатики: від камінчиків, вузликів і кісток до сучасних оптоелектронних систем. Сучасна інформатика має дещо ширше значення за визначенням Н. Вінера. Основним завданням сучасної інформатики є формалізація тези канадського філософа Л. Холла: «Усе, що йде від голови, є розумним». При цьому поряд з елементарною базою значного розвитку набуло програмування. Поряд із вузькопрофільними мовами програмування (Fortran, Pascal) створені мови системного програмування C, C^{++} і міжієрархічного програмування (Python). Значно змінилася і структура комп'ютерних наук. Подальший розвиток комп'ютерних систем, очевидно, пов'язаний зі скороченням часу і спрощенням процедури отримання необхідної інформації. Також обговорюються можливі шляхи її реалізації.

Ключові слова: синтез, кібернетика, інформатика, штучний інтелект, системний аналіз, поліметричний аналіз, принцип Моісеєва, Пайтон.

Introduction

In the main problems of evolution the cybernetics and computer science wu must select next aspects: problems of calculation (computation; problem of organization this calculations and area

https://doi.org/10.32782/mathematical-modelling/2023-6-1-1

of possible applicaqtions these calculations) [1–23]. These three aspects are determined by the level of development of civilization at one or another period of its existence. It should be noted that these three aspects are characteristic of all epochs to one degree or another. Therefore, we will analyze the evolution of cybernetics and informatics based on this point of view.

Short historical analysis of this problem is representiung. In is including Ancient Egyptian, Sumerian, Indian, Chinese, Antiquity, Jewish, Peru and other civilizations, including mythology, Pythagorian, Plato and other systems, Greek abacus, the Peruvian system of nodal counting [2; 16; 23].

The role of Blaise Pascal and Wilhelm Leitzbnitz in establishing the foundations of computer science is noted [2; 12–16; 23].

The next stage in the development of computer science was the research of Charles Babbage and Lady Ada Lovelace [2; 3; 12–16; 23]. Ada Lovelace was initiating the programming procedure.

The concept of cybernetics as the management of ships originated in Greece [2; 16; 23]. In the 19th century, it was formulated as a science of management by F.-M. Ampere [2; 16; 23] and B. Trentowski [17]. It was completed by N. Wiener, according to whom cybernetics is the science of control in the living and non-living world [11]. Later, cybernetics became the basis of computing. In its bowels, the theory of automatic regulation was expanded and the foundations of modern information theory were formulated. As F. George showed, cybernetics is a synthetic science that includes a number of sciences that are needed to solve the relevant problems [4]. Research has been conducted on the development of the hardware base of modern cybernetics and computer science: from pebbles, nodules and bones to modern optoelectronic systems.

Modern computer science has a somewhat broader meaning as defined by N. Wiener [12–15]. The main task of modern computer science is the formalization of the thesis of the Canadian philosopher L. Hall "Everything that comes from the head is intelligent" [1; 2; 16]. In this case, along with the elementary base, programming received significant development. Along with narrow-profile programming languages (Fortran, Pascal), the system programming languages C and cross-hierarchical programming (Python [9; 10]) have been created [11–15].

The structure of computer science has also changed significantly. The further development of computer systems is obviously related to the reduction of time and simplification of the procedure for obtaining the necessary information.

Possible ways of implementing this are also discussed.

Main results

The history of computer science began long before the modern discipline of computer science, usually appearing in forms like mathematics or physics. Developments in previous centuries alluded to the discipline that we now know as computer science [12–16]. This progression, from mechanical inventions and mathematical theories towards modern computer concepts and machines, led to the development of a major academic field, massive technological advancement across the Western world, and the basis of a massive worldwide trade and culture.

It is certain that the analysis of the evolution of computing methods and systems should begin with ancient civilizations [2; 16].

Thus, in ancient Egypt, the elements of algebra were introduced and a universal coded system of calculations was created, which is called the tablet of the god Thoth. Based on this, the god Thoth taught the Egyptians to count, write and cultivate the Earth. The first two provisions were deciphered using the methods of modern information theory. Roughly speaking, this tablet was the prototype of the first computer [2; 16].

Spherical geometry, astrology, the basics of matrix arithmetic and the abacus were created in ancient Sumer, which was widely used in Mediterranean civilizations by both the Phoenicians and the Greeks [2; 16].

https://doi.org/10.32782/mathematical-modelling/2023-6-1-1

Matrix calculation was needed for compiling astrological tables. In addition, computational methods were needed for the development of building mechanics (Egyptian pyramids, temples) and irrigation [2; 16].

The earliest known tool for use in computation was the abacus, developed in the period between 2700 and 2300 BCE in SumerThe Sumerians' abacus consisted of a table of successive columns which delimited the successive orders of magnitude of their sexagesimal number system. Its original style of usage was by lines drawn in sand with pebbles. Abaci of a more modern design are still used as calculation tools today, such as the Chinese abacus [2; 16; 23].

In the 5th century BC in ancient India, the grammarian Pāṇini formulated the grammar of Sanskrit in 3959 rules known as the Ashtadhyayi which was highly systematized and technical. Panini used metarules, transformations and recursions [23].

The phrase "Numbers rule the world" belongs to Pythagoras. It was the Pythagorean school that connected you with numbers. In general, the Pythagorean school was a synthesis of the esoteric Egyptian ritual system and the open Sumerian and Indian ones. In 1980–83, German archaeologists excavated several cities with octagonal houses and half-eight houses in the quarter. the civilization existed in the 6th – 5th centuries BC, but does not fit into any of the known civilizations at the time. It is likely that they were the Pythagoreans [2; 16].

Plato attached great importance to mathematics. It was his classification of numbers that was the first attempt to describe existing knowledge using mathematics. According to Plato, there are three types of numbers: mathematical (pure mathematics), sensory (applied mathematics) and ideal (numerology, from a modern point of view, numerical coding of information) [2; 16].

The classifications of Aristotle and Euclid played a significant role in the development of modern science [2; 16]. Thanks to Aristotle, the classification of modern sciences began and the foundations of formal logic were developed, which was the completion of the works of Socrates and Plato. From this point of view, formal logic can be considered formalized rules for conducting a dispute. Euclid first classified mathematical disciplines and created an axiomatic method of their description, which is practically relevant in successful mathematics.

From an organizational point of view, an important role in the creation of science was played by the Alexandria Museum, which was created by a friend of Great Alexander in Alexandria, Egypt. From the 3rd century BC to the 4th century AD, it was an organization for the development of, as it is now commonly called, Hellenic culture. Here, science developed along the lines laid down by Plato and Aristotle, since Aris totel was Ptolemy's teacher [2; 16].

The Antikythera mechanism is believed to be an early mechanical analog computer. It was designed to calculate astronomical positions. It was discovered in 1901 in the Antikythera wreck off the Greek island of Antikythera, between Kythera and Crete, and has been dated to *circa* 100 BC [12–15; 23].

The traditions of the Alexandria Museum were continued in the Muslim world (Baghdad Caliphate, Merage, Morocco, Cordoba, Granada, etc.). The fact is that, according to Mohammed, every Muslim man and woman should learn from everyone, even the Chinese. In this respect, Islam was much more tolerant of science than medieval Christianity. Only later, during the Renaissance, the same traditions began to be revived in Christian countries. The reason for this was both the crusades and the cultural contacts of the Saints. And so it turned out that the Greek Alexandrian manuscripts were translated not from the Greek language, but from the Anabian language, into Latin [2; 16].

Mechanical analog computer devices appeared again a thousand years later in the medieval Islamic world and were developed by Muslim astronomers, such as the mechanical geared astrolabe by Abū Rayhān al-Bīrūnī, and the torquetum by Jabir ibn Aflah. According to Simon Singh, Muslim mathematicians also made important advances in cryptography, such as the development of cryptanalysis and frequency analysis by Alkindus. Programmable machines were also invented by Muslim engineers, such as the automatic flute player by the Banū Mūsā brothers [12–15; 23].

https://doi.org/10.32782/mathematical-modelling/2023-6-1-1

The first scientists who studied the principles of cybernetics are Su Song, Heron of Alexandria and Ctesibius [12–15; 23]. The latter invented the first artificial automatic control system – a water clock [12–15; 23].

However, further expansion of computing science and the creation of the ideological basis of modern computing were developed in Europe [2; 16].

It should be noted that the first mechanical robot that could pronounce the simplest words was constructed in the 13th century R. Bacon [2; 16].

Technological artifacts of similar complexity appeared in 14th century Europe, with mechanical astronomical clocks [23].

When John Napier discovered logarithms for computational purposes in the early 17th century, there followed a period of considerable progress by inventors and scientists in making calculating tools. In 1623 Wilhelm Schickard designed a calculating machine, but abandoned the project, when the prototype he had started building was destroyed by a fire in 1624. Around 1640, Blaise Pascal, a leading French mathematician, constructed a mechanical adding device based on a design described by Greek mathematician Hero of Alexandria. Then in 1672 Gottfried Wilhelm Leibniz invented the Stepped Reckoner which he completed in 1694 [2; 16; 23].

In the 17th century, the rules for constructing any theory were formulated (Descartes' method and Newton's four rules of inference in physics). Roughly speaking, this is the procedure of Euclidean axiomatization. Thanks to this, it became possible to realize R. Bacon's thesis that "science is as much science as there is mathematics in it" [2; 16].

Charles Babbage is often regarded as one of the first pioneers of computing [3; 23]. Beginning in the 1810s, Babbage had a vision of mechanically computing numbers and tables. Putting this into reality, Babbage designed a calculator to compute numbers up to 8 decimal points long. Continuing with the success of this idea, Babbage worked to develop a machine that could compute numbers with up to 20 decimal places [3; 23]. By the 1830s, Babbage had devised a plan to develop a machine that could use punched cards to perform arithmetical operations. The machine would store numbers in memory units, and there would be a form of sequential control. This means that one operation would be carried out before another in such a way that the machine would produce an answer and not fail. This machine was to be known as the "Analytical Engine", which was the first true representation of what is the modern computer [3; 23].

Ada Lovelace (Augusta Ada Byron) is credited as the pioneer of computer programming and is regarded as a mathematical genius [3; 23]. Lovelace began working with Charles Babbage as an assistant while Babbage was working on his "Analytical Engine", the first mechanical computer. During her work with Babbage, Ada Lovelace became the designer of the first computer algorithm, which had the ability to compute Bernoulli numbers [3; 23], although this is arguable as Charles was the first to design the difference engine and consequently its corresponding difference based algorithms, making him the first computer algorithm designer. Moreover, Lovelace's work with Babbage resulted in her prediction of future computers to not only perform mathematical calculations, but also manipulate symbols, mathematical or not. While she was never able to see the results of her work, as the "Analytical Engine" was not created in her lifetime, her efforts in later years, beginning in the 1840s, did not go unnoticed [3; 23].

Following Babbage, although at first unaware of his earlier work, was Percy Ludgate, a clerk to a corn merchant in Dublin, Ireland. He independently designed a programmable mechanical computer, which he described in a work that was published in 1909 [23].

Two other inventors, Leonardo Torres Quevedo and Vannevar Bush, also did follow on research based on Babbage's work. In his *Essays on Automatics* (1914), Torres designed an analytical electromechanical machine that was controlled by a read-only program and introduced the idea of floating-point arithmetic [11–15; 23]. In 1920, to celebrate the 100th anniversary of the invention of the arithmometer, he presented in Paris the Electromechanical Arithmometer, which consisted

https://doi.org/10.32782/mathematical-modelling/2023-6-1-1

of an arithmetic unit connected to a (possibly remote) typewriter, on which commands could be typed and the results printed automatically. Bush's paper *Instrumental Analysis* (1936) discussed using existing IBM punch card machines to implement Babbage's design [11–15; 23]. In the same year he started the Rapid Arithmetical Machine project to investigate the problems of constructing an electronic digital computer [11–15; 23].

Up to and during the 1930s, electrical engineers were able to build electronic circuits to solve mathematical and logic problems, but most did so in an *ad hoc* manner, lacking any theoretical rigor. This changed with switching circuit theory in the 1930s. From 1934 to 1936, Akira Nakashima, Claude Shannon, and Viktor Shetakov published a series of papers showing that the two-valued Boolean algebra, can describe the operation of switching circuits [11–15; 23]. This concept, of utilizing the properties of electrical switches to do logic, is the basic concept that underlies all electronic digital computers. Switching circuit theory provided the mathematical foundations and tools for digital system design in almost all areas of modern technology [11–15; 23].

While taking an undergraduate philosophy class, Shannon had been exposed to Boole's work, and recognized that it could be used to arrange electromechanical relays (then used in telephone routing switches) to solve logic problems. His thesis became the foundation of practical digital circuit design when it became widely known among the electrical engineering community during and after World War II [11–15; 23].

Since the values stored by digital machines were not bound to physical properties like analog devices, a logical computer, based on digital equipment, was able to do anything that could be described "purely mechanical". The theoretical Turing Machine, created by Alan Turing, is a hypothetical device theorized in order to study the properties of such hardware [11–15; 23].

The mathematical foundations of modern computer science began to be laid by Kurt Gödel with his incompleteness theorem (1931). In this theorem, he showed that there were limits to what could be proved and disproved within a formal system. This led to work by Gödel and others to define and describe these formal systems, including concepts such as mu-recursive functions and lambda-definable functions [11–15; 23].

In 1936 Alan Turing and Alonzo Church independently, and also together, introduced the formalization of an algorithm, with limits on what can be computed, and a "purely mechanical" model for computing. This became the Church-Turing thesis, a hypothesis about the nature of mechanical calculation devices, such as electronic computers. The thesis states that any calculation that is possible can be performed by an algorithm running on a computer, provided that sufficient time and storage space are available [11–15; 23].

In 1936, Alan Turing also published his seminal work on the Turing machines, an abstract digital computing machine which is now simply referred to as the Universal Turing machine. This machine invented the principle of the modern computer and was the birthplace of the stored program concept that almost all modern day computers use [11–15; 23]. These hypothetical machines were designed to formally determine, mathematically, what can be computed, taking into account limitations on computing ability. If a Turing machine can complete the task, it is considered Turing computable.

The Los Alamos physicist Stanley Frankel, has described John von Neumann's view of the fundamental importance of Turing's 1936 paper, in a letter: I know that in or about 1943 or '44 von Neumann was well aware of the fundamental importance of Turing's paper of 1936… Von Neumann introduced me to that paper and at his urging I studied it with care. Many people have acclaimed von Neumann as the "father of the computer" (in a modern sense of the term) but I am sure that he would never have made that mistake himself. He might well be called the midwife, perhaps, but he firmly emphasized to me, and to others I am sure, that the fundamental conception is owing to Turing [23].

John V. Atanasoff (1903–1995) created the first electric digital computer, known as the Atanasoff-Berry computer [11–15; 23].

https://doi.org/10.32782/mathematical-modelling/2023-6-1-1

The world's first electronic digital computer, the Atanasoff–Berry computer, was built on the Iowa State campus from 1939 through 1942 by John V. Atanasoff, a professor of physics and mathematics, and Clifford Berry, an engineering graduate student [11–15; 23].

In 1941, Konrad Zuse developed the world's first functional program-controlled computer, the Z3. In 1998, it was shown to be Turing-complete in principle. Zuse also developed the S2 computing machine, considered the first process control computer. He founded one of the earliest computer businesses in 1941, producing the Z4, which became the world's first commercial computer. In 1946, he designed the first high-level programming language, Plankalkül. Now progrming languaages are more complex [11–15; 23]. They eveloped in next way – Algol – Fortran – Pascal – C C^{++} – Python [9; 10; 13].

In 1948, the Manchester Baby was completed; it was the world's first electronic digital computer that ran programs stored in its memory, like almost all modern computers. The influence on Max Newman of Turing's seminal 1936 paper on the Turing Machines and of his logico-mathematical contributions to the project, were both crucial to the successful development of the Baby [11–15; 23].

In 1950, Britain's National Physical Laboratory completed Pilot ACE, a small scale programmable computer, based on Turing's philosophy. With an operating speed of 1 MHz, the Pilot Model ACE was for some time the fastest computer in the world[52][57]. Turing's design for ACE had much in common with today's RISC architectures and it called for a high-speed memory of roughly the same capacity as an early Macintosh computer, which was enormous by the standards of his day. Had Turing's ACE been built as planned and in full, it would have been in a different league from the other early computers [11–15; 23].

Claude Shannon helped creating the foundation of information theory [2; 16].

The first actual computer bug was a moth. It was stuck in between the relays on the Harvard Mark II [23]. While the invention of the term 'bug' is often but erroneously attributed to Grace Hopper, a future rear admiral in the U.S. Navy, who supposedly logged the "bug" on September 9, 1945, most other accounts conflict at least with these details. According to these accounts, the actual date was September 9, 1947 when operators filed this 'incident' – along with the insect and the notation "First actual case of bug being found" (see software bug for details).

Cybernetics as a science has combined such areas as: control system, biology, neurology, mechanical engineering, etc. Engineer Harold Black gave rise to electronic control systems with his famous works. In 1927, his works were published, which described how, using negative feedback, amplifiers can be controlled. It was later used in artillery and radar control schemes during World War II [2; 16; 23].

In the form in which it is known today, cybernetics began to take shape in 1940. This was due to the work of such prominent scientists as W. Ashby, W. Walter, McCulloch and Wiener. We should not forget about John von Neumann, a scientist who became famous for his work in the field of mathematics and computer science. He made the most important addition – he introduced such a concept as a cellular atom and its self-reproduction. The father of cybernetics is called the scientist from the USA N. Wiener. In 1948 he published a book called Cybernetics [2; 11; 16; 23].

Rough chema of Cybernetics as synthetic science is represented in Fig. 1 [5; 11].

Really, this synthesis may be change. Now we have physical cybernetics [19], economical cybernetyics, etc. [2; 16].

The term artificial intelligence was credited by John McCarthy to explain the research that they were doing for a proposal for the Dartmouth Summer Research [8]. The naming of artificial intelligence also led to the birth of a new field in computer science. On August 31, 1955, a research project was proposed consisting of John McCarthy, Marvin L. Minsky, Nathaniel Rochester, and Claude E. Shannon [8]. The official project began in 1956 that consisted of several significant parts they felt would help them better understand artificial intelligence's makeup.

McCarthy and his colleagues' ideas behind automatic computers was while a machine is capable of completing a task, then the same should be confirmed with a computer by compiling a program to

https://doi.org/10.32782/mathematical-modelling/2023-6-1-1

perform the desired results. They also discovered that the human brain was too complex to replicate, not by the machine itself but by the program. The knowledge to produce a program that sophisticated was not there yet [8].

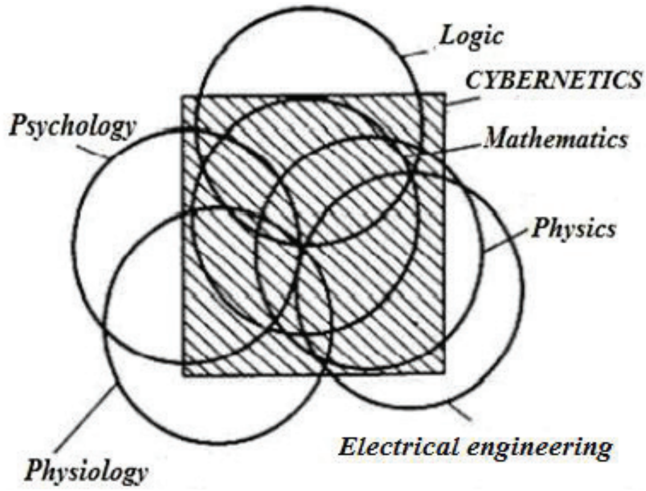

Fig. 1. A schem that roughly illustrates the areas of intersection of the main disciplines that feed cybernetics [4; 11]

The concept behind this was looking at how humans understand our own language and structure of how we form sentences, giving different meaning and rule sets and comparing them to a machine process. The way computers can understand is at a hardware level. This language is written in binary (1s and 0’s). This has to be written in a specific format that gives the computer the ruleset to run a particular hardware piece [8].

Minsky’s process determined how these artificial neural networks could be arranged to have similar qualities to the human brain. However, he could only produce partial results and needed to further the research into this idea [8].

McCarthy and Shannon’s idea behind this theory was to develop a way to use complex problems to determine and measure the machine’s efficiency through mathematical theory and computations. However, they were only to receive partial test results [8].

The idea behind self-improvement is how a machine would use self-modifying code to make itself smarter. This would allow for a machine to grow in intelligence and increase calculation speeds. The group believed they could study this if a machine could improve upon the process of completing a task in the abstractions part of their research.

The group thought that research in this category could be broken down into smaller groups. This would consist of sensory and other forms of information about artificial intelligence. Abstractions in computer science can refer to mathematics and programming language.

Their idea of computational creativity is how the program or a machine can be seen in having similar ways of human thinking. They wanted to see if a machine could take a piece of incomplete information and improve upon it to fill [8].

Logic played a significant role in the development of computational sciences. Probably precisely because of the fact that W. Leibniz tried to build a universal calculus with the help of formal logic [2; 16]. Already in the 19th century, Boolean logic was created, which was introduced into computer science as a binary number system. Later, Russell’s inductive logic types were developed, which made it possible to describe more complex systems with the help of logic [2; 16]. In the early stages of the development of computer technology, this was enough. The pinnacle of this line of research can be considered Kleene’s metamathematics, in which the theory of recursive functions was developed. Russell and Whitehead put logic in the foundations of mathematics. However, Gödel’s incompleteness theorems showed the futility of this approach. In the end, it turned out that no single

https://doi.org/10.32782/mathematical-modelling/2023-6-1-1

derivative of modern mathematics can be the basis of all mathematics. In 1948, Whitehead abandoned this concept in favor of an organismic approach. Some believe that Gödel's incompleteness theorems should be included in the theoretical foundations of computer science [2; 16].

The evolution of general theoretical approaches in computer science can be traced on the basis of S. Kleene [20] and S. Wolfram's [21] Metamathematics. If in the first book the main emphasis is on logical doctrine, then in the second book on a group of mathematical disciplines that are basic in computer programs. And that approach is intuitionistic too. However, S. Wolfram points out that it is necessary to bring computer science closer to physics [21].

However, at the current stage of the development of science, this was not enough. The need to shift mathematics to physics was expressed by many people, and even physical cybernetics arose in this way. This leads to the conclusion that it is not worth returning to the Euclid-Newton line in computer science, taking into account all the pros and cons of previous research [2; 16].

We can talk about the convergence of information theory and physics on the basis of the generalized de Broglie ratio [2, 16, 22]

$$\frac{S_a}{\hbar} = \frac{S_e}{k_B} = S_g, \quad (1)$$

about the equality of ordered and disordered information in closed system. Here S_a is an action, S_e – entropy, $\hbar$ – Planck constant, k_B Boltzmann constant [5; 7; 8].Therefore, it makes sense to consider dimensionless relations not as elements of dimensionless entropy or action, but as elements of a generalized information [2; 16; 22].

This ratio is valid for closed systems. For open systems, we can formulate the following principle [2; 16]:

$$\delta S_g > 0. \quad (2)$$

If $S_g = S_e$ we have Shennon law of information theory and Klimontovich criterion of open systems [2; 16].

In addition, relation (1) is the rationale for the introduction of information numerosity (simple, technical and generalized) as the quantifiers of relevant calculation operations, cells as we can see from (1) include relevant quanta of physical processes.

A theory, which include physical and other processes in procedure of computation, can be a polymetric analysis – a theory of variable measure or a theory with a variable hierarchy [1; 2; 16]. The main components of this theory are: functional numbers (generalization of quadratic forms); generalized mathematical transformations (quantitative and qualitative, responsible for single-hierarchical and multi-hierarchical calculations), information grids built on generalized constructive elements (algebraic combination of functional numbers and generalized mathematical transformations). the theory of information calculations, based on the principle of optimal information calculations (mathematical forms may be various, including (2)), the polymetric theory of measure and measurements (synthesis of the theories of measure and analysis of dimensions into a single system) and the hybrid theory of systems (practically systems of possible formalization of knowledge). The theory itself is based on two criteria: the criterion of completeness (the criterion of the layout of the system based on its completeness) and the criterion of simplicity (the criterion of the optimality of this layout, which includes the criterion of the optimal amount of information) [1; 2; 16].

Since the basis of polymetrics is the completeness of the system in a more general sense, as in K. Gödel, and in addition, it is connected with the complexity of the system, then of course the polymetric method can be laid at the basis of computer science. With the help of the hybrid theory of systems, it was possible solve the S. Beer centurial problem in cybernetics (the problem of information complexity). Unlike Russell's inductive logical types, the number of which can be arbitrarily large. in the hybrid theory of systems, the number of types is finite, but the number of implementations can be arbitrarily large [1; 2; 11; 16].

https://doi.org/10.32782/mathematical-modelling/2023-6-1-1

In an epistemological sense, Gödelian completeness and Occam's edge are equivalent to the fourth rule of the Newtonian four rules of inference in physics. However, unlike logic, Newton's fourth law can be applied to almost all sciences without exception. Thus, in computer science, we can move from Leibniz's concept to the more general Euclid-Newton [2; 16].

However, since the polymetric analysis has a variable hierarchy, it is possible to create any number of Kleene or Wolfram-type systems based on it. The approaches of Kleene and Wolfram are inductive, and polymetric analysis is deductive. Therefore, this method can be considered as a deductification of intuitionism as well.

In whole Polymetrical Analysis and possible computer sciences must satisfy six conditionswith point of conditions, which are formulated for the general theories (theories of everything) [1; 2; 11; 16]:

1. It must be open theory or theory with variable hierarchy.
2. This theory must be having minimal number of principles.
3. It must based on nature of mathematics (analysis, synthesis and formalization all possible knowledge).
4. We must create sign structure, which unite verbal and nonverbal knowledge (mathematical and other) in one system.
5. We must have system, which is expert system of existing system of knowledge and may be use for the creation new systems of knowledge.
6. Principle of continuity must be true for all science.

These conditions must be used for the creation any dynamic science, which can be presented as open system.

Conclusions

The problems of evolution in cybernetics and computer sciences are analyzed. Short histarical analysis this problem in ancient civilizations is observing.

Role antiquity and middle age scientists in the development of computer science is showing. Role of B. Pascal, W. Luibniz, Ch. Babbage and A. Lovelace researches in the development of computer science is discussing.

Development of logica line of cybernetics and computer science and its limitations is observing. Evolution of inuitionism line in computer science on the example Kleene and Wolfram systems is representing. Short analysis Cybernetics as synthetic system is giving.

We show the necessary of search more universal concept of computer science. Thi concept is basin on six rules. As example of this concept, polymetric analysis is representing.

Bibliography

1. Trokhimchuck P.P.S. Beer centurial problem in cybernetics and methods of its resolution. *Advanced in computer science* / ed. Mukesh Singla. New Delhi : AkiNik Publications, 2020. Vol. 7. Ch. 5. P. 87–118.
2. Trokhimchuck P.P. Theories of Everythings: Past, Present, Future. Saarbrukken : Lambert Academic Publishing, 2021. 260 p.
3. George F.H. Philosophical Foundations of Cybernetics. London : Abacus Press, 1976. 157 p.
4. George F.H. Foundations of cybernetics. London, Paris, New York : Gordon and Breach Science Publishing 1977. 260 p.
5. Nillson N.I. The Quest for Artificial Intelligence: A History of Ideas and Achievements. New York : Cambridge University Press, 2010. 562 p.
6. Computer science. *Wilkipedia*. URL: http:/en.wikipedia.org/wiki/ Computer_ science
7. Computing. *Wilkipedia*. URL: http:/en.wikipedia.org/wiki/Computing
8. Nillson N.I. Artificial Intelligence. A new synthesis. San Francisco : Morgan Kauffman Publishers, Inc., 1998. 514 p.

https://doi.org/10.32782/mathematical-modelling/2023-6-1-1

9. Замуруєва О.В., Кримусь А.С., Ольхова Н.В. Об'єктно-орієнтоване програмування в Python : курс лекцій. Луцьк : Вежа-Друк, 2018. Частина 1. 64 с.
10. Замуруєва О.В., Вілігурський О.М. Об'єктно-орієнтоване програмування в Python : курс лекцій. Луцьк : Вежа-Друк, 2020. Частина 2. 72 с.
11. Trokhimchuck P., Viligurskii O., Zamurueva O. Some questions of synthei in cybernetics and computer science. *Applied Problems of Mathematical Modelling.* 2022. Vol. 5. Ch. 2. P. 84–89.
12. Ifrah G. The Universal History of Computing: From the Abacus to the Quantum Computer. New York : John Wiley & Sons, 2021. 416 p.
13. Koetsier T. On the prehistory of programmable machines: musical automata, looms, calculators. *Mechanism and Machine Theory.* 2001. Vol. 36. Is. 5. P. 589–603.
14. Rendall B. From Analytical Engine to Electronic Digital Computer: The Contributions of Ludgate, Torres, and Bush. *Annals of the History of Computing.* 1982. Vol. 4. Is. 4. 20 p.
15. O'Regan G. Introduction to the History of Computing: A Computing History Primer (Undergraduate Topics in Computer Science). Berlin : Springer Verlag, 2016. 317 p.
16. Трохимчук П.П. Математичні основи знань. Поліметричний підхід. 2-е вид. Луцьк : Вежа-Друк, 2014. 624 с.
17. Trentowski B. tounek filozofii do cybernetyki czyli sztuka rzadzenia narodem. Poznan, 1843. 196 s.
18. Кухтенко А.И. Кибернетика и фундаментальные науки. Киев : Наукова Думка, 1987. 144 с.
19. Самойленко Ю.И. Проблемы и методы физической кибернетики. Киев : Ин-т математики НАН Украины, 2006. 644 с.
20. Kleene S.C. Introduction to metamathematics. Amsterdam : North Holland, 1965. 550 p.
21. Wolfram S. Metamathematics: Foundations & Physicalization. Leipzig : Wolfram Media, 2022. 456 p.
22. De Broglie L. Thermodynamics of Isolated Point. The hidden thermodynamics of particles. Paris : Gauthier Villars, 1964. 98 p.
23. History of computer science. *Wikipedia*. URL: https://en.wikipedia.org/wiki/History_of_computer_science

References

1. Trokhimchuck, P.P.S. (2020). Beer centurial problem in cybernetics and methods of its resolution. *Advanced in computer science* / ed. Mukesh Singla. Vol. 7. Ch. 5. New Delhi : AkiNik Publications. P. 87–118.
2. Trokhimchuck, P.P. (2021). Theories of Everythings: Past, Present, Future. Saarbrukken : Lambert Academic Publishing. 260 p.
3. George, F.H. (1976). Philosophical Foundations of Cybernetics. London : Abacus Press. 157 p.
4. George, F.H. (1977). Foundations of cybernetics. London, Paris, New York : Gordon and Breach Science Publishing. 260 p.
5. Nillson, N.I. (2010). The Quest for Artificial Intelligence: A History of Ideas and Achievements. New York : Cambridge University Press. 562 p.
6. Computer science. *Wikipedia*. URL: http:/en.wikipedia.org/wiki/ Computer_ science
7. Computing. *Wikipedia*. URL: http:/en.wikipedia. org / wiki/ Computing
8. Nillson, N.I. (1998). Artificial Intelligence. A new synthesis. San Francisco : Morgan Kauffman Publishers, Inc. 514 p.
9. Zamuruyeva, O.V., Krymus, A.S. & Olkhova, N.V. (2018). Ob'iektno-oriientovane prohramuvannia v Python – [Object-oriented programming in Python] : Course of lectures. Part 1. Lutsk : Vezha-Druk. 64 p. [in Ukrainian]
10. Zamurueva, O.V. & Viligurskyi, O.M. (2020). Ob'iektno-oriientovane prohramuvannia v Python – [Object-oriented programming in Python] : Course of lectures. Part 2. Lutsk : Vezha-Druk. 72 p. [In Ukrainian]

https://doi.org/10.32782/mathematical-modelling/2023-6-1-1

11. Trokhimchuck, P., Viligurskii, O. & Zamurueva, O. (2022). Some questions of synthei in cybernetics and computer science. *Applied Problems of Mathematical Modelling*. Vol. 5. Ch. 2. P. 84–89.
12. Ifrah, G. (2021). The Universal History of Computing: From the Abacus to the Quantum Computer. New York : John Wiley & Sons. 416 p.
13. Koetsier, T. (2001). On the prehistory of programmable machines: musical automata, looms, calculators. *Mechanism and Machine Theory*. Vol. 36. Is. 5. P. 589–603.
14. Rendall, B. (1982). From Analytical Engine to Electronic Digital Computer: The Contributions of Ludgate, Torres, and Bush. *Annals of the History of Computing*. Vol. 4. Is. 4. 20 p.
15. O'Regan, G. (2016). Introduction to the History of Computing: A Computing History Primer (Undergraduate Topics in Computer Science). Berlin : Springer Verlag. 317 p.
16. Trokhimchuck, P.P. (2014). Matematychni osnovy znan. Polimetrychnyi pidkhid – [Mathematical foundations of knowledge. Polymetric approach]. 2nd edition. Lutsk : Vezha-Druk, 624 p.
17. Trentowski, B. (1843). The attitude of philosophy to cybernetics, or the art of governing a nation. Poznan. 196 p.
18. Kukhtenko, A.I. (1987). Kybernetyka y fundamentalni nauky – [Cybernetics and Fundamental Sciences]. Kyiv : Naukova Dumka. 144 p.
19. Samoylenko, Yu.I. (2006). Problemy y metody fyzycheskoi kybernetyky – [Problems and methods of physical cybernetics]. Kyiv : Institute of Mathematics of the National Academy of Sciences of Ukraine, 644 p.
20. Kleene, S.C. (1965). Introduction to metamathematics. Amsterdam : North Holland. 550 p.
21. Wolfram, S. (2022). Metamathematics: Foundations & Physicalization. Leipzig : Wolfram Media. 456 p.
22. De Broglie, L. (1964) Thermodynamique du point isolé. La thermodynamique cachée des particules. Paris : Gauthier Villars. 98 p.
23. History of computer science. *Wikipedia*. URL: https://en.wikipedia.org/wiki/History_of_computer_science

Trokhymchuk Petro Pavlovych – PhD, Associate Professor, Associate Professor at Anatoliiy Svidzinskyi Department of Theoretical and Computer Physics of Lesya Ukrainka Volyn National University, e-mail: trope1650@gmail.com, ORCID: 0000-0003-2737-0506.

Vilihurskyi Oleh Mykolaiovych – Senior Lecturer at Anatoliiy Svidzinskyi Department of Theoretical and Computer Physics of Lesya Ukrainka Volyn National University, e-mail: viligurskyi@gmail.com.

Zamuruieva Oksana Mykolaivna – PhD, Associate Professor, Associate Professor at Anatoliiy Svidzinskyi Department of Theoretical and Computer Physics of Lesya Ukrainka Volyn National University, e-mail: zamurueva@gmail.com, ORCID: 0000-0003-0032-0613.

Трохимчук Петро Павлович – к.ф.-м.н., доцент, доцент кафедри теоретичної та комп'ютерної фізики імені А.В. Свідзинського Волинського національного університету імені Лесі Українки, e-mail: trope1650@gmail.com, ORCID: 0000-0003-2737-0506.

Вілігурський Олег Миколайович – старший викладач кафедри теоретичної та комп'ютерної фізики імені А.В. Свідзинського Волинського національного університету імені Лесі Українки, e-mail: viligurskyi@gmail.com.

Замуруєва Оксана Миколаївна – к.ф.-м.н., доцент, доцент кафедри теоретичної та комп'ютерної фізики імені А.В. Свідзинського Волинського національного університету імені Лесі Українки, e-mail: zamurueva@gmail.com, ORCID: 0000-0003-0032-0613.

https://doi.org/10.32782/mathematical-modelling/2023-6-1-1