

The Nature of Computational Thinking Used in Educational Research

Rujuan Wang

College of Humanities & Sciences
Northeast Normal University
Changchun, China
wangrujuan_1108@sina.com

Lei Wang*

College of Humanities & Sciences
Northeast Normal University
Changchun, China
36334157@qq.com

Abstract—The cultivation of computational thinking is not a requirement of computer science, but an important part of general education in the whole university. Thinking cannot be restricted by professional. Students of different professional backgrounds have common characteristics in thinking problems, solving problems and dealing with problems. This paper discusses two unique properties of computational thinking, which can be used to explain and relate the world. These two qualities are the boundaries between computational thinking and empirical thinking and logical thinking. This paper theoretically expounds the connotation of these two characteristics and their position in computational thinking. At the same time, the significance and function of these two characteristics to computer science and computer engineering are further discussed, especially in Computer Engineering, which become the important theoretical basis and background, and influence the quality standards and development standards of engineering technology. Combined with the practice of teaching reform, this article puts forward that we should strengthen the understanding of this aspect in teaching contents.

Keywords—Computational thinking; proof of explanation; education; Association of world

I. INTRODUCTION

Computational thinking is a thinking mode in which human beings participate in the process of thinking and solve problems. In this way, information processing can be carried out quickly and effectively, and the solution to the problem is Cuny et al. [1] proposed. As expected, in the study, Papert [2] proposed procedural thinking (i.e., "thinking like a computer") is considered to be part of the overall thinking skill. Until 2006, Jeanette Wing in the ACM American computer society communication published "computational thinking" this article, since then, the calculation of thinking has been a new definition. Wing's article suggests that computational thinking is not only a useful skill for experts in computer science, but a mental process that anyone can use to solve problems and discover computational solutions. In this broader sense, considered as part of logical thinking, computational thinking can be seen as a skill related to all disciplines, not just computer science. Therefore, Wing [3] argued: "to reading, writing and arithmetic, we should add computational thinking to every child's analytical ability".

This paper attempts to discuss the core concepts of computational thinking, the basic question is whether

computational thinking have their own unique thinking and judgment standards, which are not recognized in the empirical and logical thinking framework. This article is divided into four parts. In the first part, the paper give a brief description of the classification standards of scientific thinking; in the second part, review what is the standard of the physical and logical thinking as an empirical thinking, what is the essential difference in judging the correctness of the conclusion; in the third part, we introduce the characteristics of computational thinking; finally, we make some summaries.

II. CLASSIFICATION AND CHARACTERISTICS OF SCIENTIFIC THINKING

Scientific thinking is the thinking in scientific activities. Generally speaking, we are categorized by the external form of thinking, which is mainly the form of thinking and the criterion for judging the correctness of the conclusion. The purpose of scientific thinking is to produce conclusions, so how to produce conclusions and accept conclusions by what standard constitutes a unique symbol of thinking, and the main basis for different modes of thinking.

For example, in the model of positive thinking, people get some conclusions that reveal the objective world through observation and experiment. The most important part of these conclusions is in the form of law, while other conclusions appear in the way of logical inference or experimental verification. For the same objective world, a variety of theoretical systems can be set up to explain, so people must agree a kind of interpretation to develop the understanding and description of the laws of the objective world. This is the physical science system established by scientists such as Galileo, Newton and Einstein. In this system, we start from a few laws and set up a framework to explain the whole objective world. The most significant feature is that any conclusion, regardless of the way it is obtained, is not recognized as long as it is not observed or verified in the experiment. Positivism is the sole criterion for determining the correctness of conclusions, which is the specificity of empirical thinking.

In logic thinking, it is entirely another way to produce conclusions and determine the correctness of conclusions. First, there is a set of propositions called axiom, and then there is a rule of reasoning. The conclusion from the reasoning rules from the axiom is considered correct. No matter how absurd

the conclusion sounds. For some conclusions, although it may be repeatedly verified in the real world, it will not be considered the correct conclusion as long as it cannot be derived from this system by reasoning. Logical reasoning is the sole criterion for judging a conclusion, which is the particularity of logical thinking.

These two modes of thinking obviously have very different characteristics, so they become the two modes of thinking in scientific thinking; their representatives are physics and mathematics. We do not rule out that we can build scientific systems in other ways, only because of the historical development of human society and the adoption and acceptance of the present system of expression on physics and mathematics. Therefore, we must follow the principles together, and the conclusion that conforms to the criteria must be considered correct, and the conclusion that does not conform to the criteria of judgment can only be doubted. This is the basic scientific contract of human society. Science can develop in a healthy and orderly way by observing this contract only.

Based on this principle, it can explain why there is no chemical thinking and biological thinking, nor economic thinking and management thinking. Different disciplines can adopt the same way of thinking, so long as the way of producing conclusions and their criteria, their thinking mode is the same. In chemistry or economics, the way and standard of thinking is not different from physics or mathematics, or just the mixture of them. The difference lies in the different objects of thinking, and the way of thinking conclusion is the same as the judging standard. The classification of thinking patterns is not based on the object of thinking. Some subjects in the humanities, such as literature, art, etc., contain the characteristics of artistic thinking, which are not entirely the category of scientific thinking. There are other requirements for the conclusions and criteria in these thinking.

Therefore, as computational thinking, there should be a unique way of producing conclusions and criteria so that we can become a new mode of thinking. According to Zhou's view, the most essential characteristic of calculation thinking is abstraction and automation, and its expression forms have some characteristics such as limited, procedural, mechanistic, and feasible. However, this statement still does not completely remove the computational thinking and logical thinking. It is too simple to elaborate on the concept of abstraction and automation, and it is difficult to expand into a theoretical system. In fact, abstraction is also a characteristic of mathematics. Mathematics itself also acknowledges constructional, implementation, and iteration. Therefore, only from these aspects to explain the characteristics of calculation thinking, although grasp the essence, but the lack of a certain theoretical basis and characterization, it seems not enough to disconnect computer science from mathematics, so we need to explore the more essential internal capacity behind the calculation thinking and completely different from the positive thinking and logic. That is to say, what kind of view and standard of Computational Thinking provide a view of the world, and explore the world view in the field of computational thinking [4].

III. THE NATURE OF COMPUTATIONAL THINKING

The most essential characteristic of computational thinking is abstraction and automation. Its form of expression is limited, procedural, mechanical and feasible. However, this statement still does not completely remove the computational thinking and logical thinking, only from the concept of abstraction and automation, which is too simple to expand into a theoretical system. In fact, abstraction is also a characteristic of mathematics. Mathematics itself also acknowledges constructional, implementation, and iterative iteration. Therefore, only from these aspects to explain the characteristics of calculation thinking, although grasp the essence, but the lack of a certain theoretical basis and characterization, it seems not enough to disconnect computer science from mathematics, so we need to explore the more essential internal capacity behind the calculation thinking and completely different from the positive thinking and logic.

The computer science system also has its basic assumptions, which constitute the cornerstone of computer science. In this paper, we propose three important cornerstones of computer science, as flow: Turing machine theory; feasible computability; Iterative computability.

A. *The Nature of Computational Thinking--Explainable Proof*

Interactive authentication system is a new way of judging conclusions, which is achieved through information interaction between Verifier and Prover. P and V gradually achieved the recognition of certification during the interaction process. Since V is validated in a definite way, the conclusion is reliable. P provides evidence in a way of "oracle", though the ability is unlimited, but without the verification of V, the proof process cannot be ended. This kind of proof is like in the classroom, the student and teacher's interaction, the student is equivalent to the V, the teacher is equivalent to the P, the student constantly asks questions to the teacher, and the teacher always can answer the student's question correctly, finally makes the student believe the conclusion. Therefore, this method is called "interpretable proof" in document [5]. This name illustrates the characteristics of interactive proof vividly. Interpretable proof is a new way of proving different from mathematics. We call it computable provability.

A mathematical proof is a one-time proof that the proof of a mathematical problem can be written off once, and anyone can believe the conclusion by reading the proof later. In order to limit this problem to a reasonable range, we generally equate the class of mathematical provable problems to the class of NP problems. A NP problem, if the solution can be found, proves that the length of the solution will not be too long, so it can be written down, so the NP problem class is a writable proof class. From this point of view, the problem class that can be written proof is NP, and it can be explained that the class of proof problem is IP. So, can the class of written proof problem (i.e., mathematical prove) be equal to the class of interpretable proof problem. This is a question that does not know the answer at the moment. If IP is NP, mathematically provable cannot cover computable provability. If $IP=NP$, computing science will collapse into mathematics.

The proof of one interaction is mathematical proof, and the proof that must be repeated many times does not conform to the mathematical standard. So IP has gone beyond the mathematical provable range. If a NP problem is not proved, finding a proof is only a matter of time. Once a proof is found, it can be written down once and for all. But an IP problem will experience a different process of proof in different situations. These processes may not be repeatable and consistent, but they are only a way of proof. The interpretable proof of a problem itself cannot be mathematically proved unless $IP=NP$. Naturally, it is impossible to write down all the questions and answers in advance to prepare questions for future generations. Because for an interactive proving system with polynomial time, all possible interactive processes are exponential on scale, so it is impossible to write all of them. Interactive proofs can only be carried out according to V's questions and P's answers. This is the reason why the interpretable proof itself cannot be mathematically proved.

The interactive system emphasizes that the solution of the problem is realized in the interactive process, which is well reflected in the project. Whether it is in software development, system design, and problem solving, we can't expect to solve all the problems in a mathematical sense, or to prove the correctness of the software or system at one time. In most cases, this kind of mathematical proof does not exist. Therefore, in practical engineering, we require that the developed software, or the system designed, have a good response to the control object or the operating environment, that is, when the control object and the operating environment change, the control system or the running system can respond in time to ensure that the system can run correctly. Interactive systems are more suitable for the theoretical foundation of describing such engineering problems, rather than mathematics. Under the background of engineering, it is the response and adaptability of the system to environmental changes. This problem is basically unworkable with the traditional mathematical viewpoint. Therefore, the theory and application of interactive proof can solve this kind of problem well. In this sense, the correctness of software we call, the correctness of a system should be correct in the sense of interactive proof.

Since the interpretable proof itself cannot be proved by mathematics, some special methods have been adopted to illustrate the correctness of the software or system in the interpretable sense. One way is to test; testing is an interaction, taking a pre designed test program, checking possible problems to determine the ability of the software or system to respond to changes. Testing is not an exhaustive method, so it is impossible to cover all possible problems, but just want the content of the test to be representative and reflect the possible problems as much as possible. There are various testing theories and practical methods. In practical applications, the establishment of benchmark is a common way. Through the test platform, there is an objective comparison between different products. In addition to testing the performance evaluation of the product itself, it also provides the performance comparison between the same types of different products.

Interaction and interactive proofs are the awareness that we should focus on in computer courses. This training was not enough before. Because of the profound influence of mathematics, in many courses of computer, it is often used to consider solving problems and designing systems with mathematical ideas and standards. These methods are not effective on many engineering problems. The reason is that computer science or computer engineering and mathematics are two different ways of thinking. Solving engineering problems with mathematical thinking way will make our thinking too narrow and absolute, and it is difficult to cultivate real computer science and engineering application talents. Another extreme is that, because engineering problems are difficult to be evaluated by mathematical methods, in the design and development of many engineering problems, there is no theoretical guidance in the design and development of many engineering problems and the quality of software or products is not strictly guaranteed. These two situations exist now, and the latter seems to be more common, so it is a suitable choice to introduce the theory of interactive proof system into engineering design and development. This is one of the revelations that computational thinking gives us completely different from logical thinking, and one of the issues that we should focus on in promoting Computational Thinking oriented computer curriculum reform.

B. The Nature of Computational Thinking--The Association of world

Another characteristic of computational thinking is the relationship. This is a way of looking at the world from the perspective of computational thinking. The world view of physics is causality, for example, if an object changes its form of motion, it must be affected by a force. And the way that mathematics looks at the world is logical relation, for example, if we make a vertical line from the vertex of the isosceles triangle to the bottom, then the vertical line must be divided equally.

Unlike physics and mathematics, the way computer science looks at the world is related, and association is a link between phenomena. For computer science, it does not care about the internal causality between phenomena, nor does it care about the logical relationship between phenomena, but only concerns the spatial and temporal relations between phenomena. The computer handles the symbol, and the symbol is the abstract of the objective object. All the phenomena are used as the data to be processed by the computer. Therefore, in the computer science, the phenomenon is the manifestation of all things in the world, and the specific content is the data. In particular, the study of data and data, which is the relationship between phenomena and phenomena, has become an important content of computer science, especially in the era of large data. These studies have also promoted the richness of the connotation of computational thinking, that is, a new world outlook that looks at the world from the perspective of relevance [6,7].

There are two main relationships among the phenomena of association research: one is spatial relationship, the relationship between the phenomena and the relationship between the phenomena; the other is the time relation, and the order relation between the phenomena is studied. In order to find out the

location correlation or time correlation from a large number of phenomena or data, some techniques have been developed to mine these associations. With regard to spatial association, the space here includes both the usual geometric space and the abstract space defined by the problem, mainly for the classification of the data, in the defined space, to divide the similar data, or to find out the clustering relation between the data. The most popular technologies are familiar support vector machines (SVM) and neural networks (NN). These two technologies are based on learning and training, and give a model to partition the data to the spatial relationship. As for time correlation, Bayesian networks and various dynamic models evolved from Bayes formula are more common at present. By modifying the model, we get the order (time) relation [8], which is close to the actual situation. The association relation is statistically significant. An event A is associated with another event B, in a statistical sense. Therefore, when A occurs, it may not necessarily happen B, but to a certain extent, it will happen B, which is the biggest difference between relevance and causality and logic. In the study of human behavior analysis, more is the relationship of the phenomenon, the qualitative or quantitative establishment and analysis of these relationships, in the social science and Humanities Science occupies an important content.

From primary school to secondary school, all of our thinking training is basically causal or logical, that is, physics and mathematics courses for our impact is too deep, and thus to develop a way of thinking that either. However, the calculation of thinking is to train us another way of thinking, this way of thinking concerned about the relationship between things, and the resulting qualitative or even quantitative analysis, this relationship has some uncertainty and variability, which Kind of thinking to broaden the perspective of the world and methods. This is the basis for solving problems using computers. Large data analysis and network science is now introduced a large number of related theory and analysis tools. Such as the scale-free theory of Barabasi [9] in the network, the theory of small-world and the power-law phenomenon in the analysis of human behavior are all important rules in association analysis. Thinking from the relationship of things to think about the problem, rather than from the perspective of causal or logical thinking, is the unique nature of the calculation of thinking, as a computer basic course, training this thinking is an important element.

IV. CONCLUSIONS

In this article, we discuss two characteristics of computational thinking, which can be used to explain the relationship between proof and relevance. This is the boundary between computational thinking and empirical thinking and logical thinking. It is also a new world view and values different from the first two kinds of thinking. The method and mode of extending some projects are related to this, laying the important position of Computational Thinking in computer science and engineering. Therefore, from the perspective of teaching, the cultivation of such thinking and the mastery of two characteristics need to be deeply explored. This is also a major change that students need to experience from high

school to university. In the field of computing thinking, college computer courses should gradually make students solve problems in the process of interaction. It is of great importance to improve the comprehensive quality of students' scientific thinking by observing the association rules between things and forming some ideas and habits of thinking problems. In practical scientific activities, all kinds of thinking are mixed, and basically do not simply use a mode of thinking to understand problems and solve problems. Therefore, the characteristics of Computational Thinking proposed in this paper are extracted as a single study, and the purpose is to clearly draw out the unique ways and standards of computational thinking, and to make these all clear, and to improve the ability to apply all kinds of thinking in the solution of practical problems. This is a point that we want to emphasize.

With the further development of network and data science, more computational thinking characteristics will be further excavated and displayed, playing a more and more important role. Of course, the specific forms involved in these two kinds of computational thinking are too special, even if the computer majors do not have to learn the content, so it is not possible to teach these contents in the basic computer course of the University, but for the teachers, we should understand the material in this respect and take the material in this field. The basic characteristics of it are to teach students to cultivate their understanding and awareness in these areas. We should pay more attention to teaching reform.

ACKNOWLEDGMENT

Jilin Province Educational Science Planning Project GH16595, GH171014, ZD16071, The Education Department of Jilin Province, "13th Five-Year" social science research plan for the year 2017 project: JJKH20170983SK, Jilin Provincial Higher Education Society 2016 Annual Higher Education Research Key Project JGJX2016B27.

REFERENCES

- [1] Cuny, J., Snyder, L. & Wing, J. (2010). Demystifying computational thinking for non-computer scientists. Work in progress.
- [2] Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. New York, NY: Basic Books.
- [3] Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33-35.
- [4] R. Penrose. (1990). *The emperor's new mind--concerning computers, minds, and the law of physics/rogerpenrose(1989)*. Kagaku, 60.
- [5] Goldwasser, Micali, & Rackoff. (1989). The knowledge complexity of interactive proof systems. *Siam Journal on Computing*, 18(1), 186-208.
- [6] Lindsay, R., & Gorayska, B. (2002). Relevance, goal management and cognitive technology. *International Journal of Cognition & Technology*, volume 1(1), 187-232.
- [7] Hjørland, B. (2010). The foundation of the concept of relevance. *Journal of the Association for Information Science & Technology*, 61(2), 217-237.
- [8] Mitchell. (2003). *Machine Learning*. McGraw-Hill.
- [9] Gelman, A. (2010). Bursts: the hidden pattern behind everything we do. *Physics Today*, 63(5), 46-46.