Intelligent Control and Automation, 2025, 16(4), 111-157
https://www.scirp.org/journal/ica
ISSN Online: 2153-0661
ISSN Print: 2153-0653

Structural Intelligence Machine Theory

Dong H. Liu

Department of Artificial Intelligence, Guangdong Polytechnic Normal University, Guangzhou, China
Email: m13926032320@163.com

How to cite this paper: Liu, D.H. (2025) Structural Intelligence Machine Theory. *Intelligent Control and Automation*, **16**, 111-157.
https://doi.org/10.4236/ica.2025.164006

Received: August 20, 2025
Accepted: September 12, 2025
Published: September 15, 2025

Abstract

This is the first paper to study the theory of structural intelligence machines. Part one summarizes the advantages of structural intelligence machines; part two introduces a number of theoretical tools for studying structural intelligence; part three proposes an innovative concept for a novel central processing unit; part four puts forward a software graph theory centered on graph thinking; part five studies the implementation and application of structural intelligence machines; and part six is a critique of current large-scale information-based models. The main purpose of the paper is to describe the benefits of structural intelligent technology for the future of human society.

Keywords

Structural, Novel Central Processing Unit, Software Diagram, Machine Life

1. Introduction to Structural Intelligence Machines

1.1. What Is a Structural Intelligence Machine?

Structural intelligence machines use equations to simulate the cells of real life, to construct equation structure diagrams to simulate the tissues of real life, and to endow the equation structure diagrams with equation unit generators and clocks. Equation unit generators simulate the intelligent activities of biological tissue, while clocks simulate the vital signs of real life. Together, they are able to simulate the cycles and heredity of real life, establishing intelligence on the basis of intelligent structural activities, allowing the equation unit generators to create new equation units, thus achieving both reproduction and heredity. The whole intelligent structure therefore consists of a logical equation structure diagram endowed with life. This diagram can be arbitrarily large, the size of a nautical chart or space chart (hereinafter referred to as a "sea chart") and created through a combination of structural hardware and software. The clock will be embedded within the central processor, with a life clock creating the vital signs of the machine.

Structural intelligence machines comprise three main components: the machine brain, the robot, and the machine society. ① Machine Brain: A novel central processing unit featuring software diagrams, where the thinking of the diagram itself it able to reflect intelligence; ② Robot: Intelligent hardware combined with software diagrams (including brain structure diagrams, intermediate structure diagrams, and basic structure diagrams), leading to a form of brain diagram thinking that controls the thinking of an organizational chart, which in turn controls the thinking of a component chart, thus demonstrating the intelligence of the machine; ③ Machine Society: A connection of various intelligent machines through a logical network to form a machine society or a human-machine society. This can utilize the thinking of the network software diagram to create social intelligence.

1.2. Alignment with the Natural Laws of Intelligent Life

The natural laws of intelligent life establish intelligence within the activities of intelligent structures, making use of their reproduction and heredity to achieve an iterative evolution of intelligence. Structural intelligence uses equations to simulate cells, and life-giving logical equation structure diagrams to simulate living tissue. Structural intelligence machines establish intelligence within the activities of the machine's equation structure diagrams, while the equation unit generator enables reproduction and heredity, thereby driving an iterative form of evolution. The clock inside the structure diagram features a life clock to create vital signs. Structural intelligence machines can therefore be aligned with the natural laws of life.

1.3. Prospects for Long-Term Iteration in Structural Intelligence Machines

1.3.1. Prospects for Long-Term Iteration

A structural intelligence approach is a path for developing intelligent machines based on the natural laws of intelligent life, using artificial intervention and machine genetics to increase the difficulty and complexity of intelligent structures. Equations reflect the associated difficulty, while the scale of the equation structure diagram reflects the required complexity. Based on this, both the difficulty and complexity of intelligent analysis and intelligent reasoning can be developed, using diagram structure analysis and its associated reasoning to achieve the desired goals. Diagram structure analysis focuses on the complex analysis created by equation difficulty, the huge number of paths within the diagram structure, and the intersections of these paths. Structure diagram reasoning focuses on deriving variables for large control variables from diagram outputs generated through complex analysis, using these variables to control both software and hardware. In addition, a clock is used to implement the vital signs of the machine, which in turn enable machine life activities and iterative evolution. A structural intelligence machine refers to a life-giving logical equation structure diagram created by a combination of software and hardware. Long-term iteration will lead to increasingly sophisticated, complex and numerous output values, strengthening the ability of these output values to control an increasingly large number of objects.

1.3.2. The Essence of Structural Intelligence Machines

The continued development of the structural intelligence approach will lead to an unprecedented level of complex analysis, reasoning and control, in stark contrast to the failure and/or redirected focus of information-based approaches. This can be achieved because structural intelligence has found the required cellular materials of the logical world—Equations. Equations can be used to simulate real living cells, and equation structure diagrams can be used to construct the life organization of all machines within the logical world. A clock can then be used to endow the structure diagram with time and vital signs, simulating reality to create machine reproduction, inheritance, iterative evolution and other life activities. The essence of structural intelligence can be used to represent the bionics of intelligent life in the universe.

1.4. Advantages of Structural Intelligence Machines

1.4.1. The Issue of Cost

The intelligence of structural machines does not require large-scale data training, meaning development costs are much lower than for information-based machines, leading to significant savings in data resources and computing power. Moreover, the equations related to structural technology have variable value sets, which can also store large-scale data. This means that structural intelligence can inherit the existing data training technology and training results obtained via information-based intelligence.

1.4.2. The Issue of Intelligence

Structural machines can perform long-path analysis of large numbers of equations on vast charts, as well as the mass analysis of massive paths, including their cross-sectional analysis. Based on this, highly complex reasoning and output for mass reasoning variables can be constructed. The potential development for this form of intelligence cannot be achieved using information-based machines, and its intelligence will far exceed that of the human brain, with the ability to simulate human social intelligence, ultimately providing the possibility of enhancing the intelligence of humanity.

1.4.3. The Issue of Vital Signs

All life in the universe has vital signs, which are of great significance for reproduction, inheritance, iterative evolution, life cycles, and other life activities. This tells us that intelligent machines must also develop vital signs. The vital signs of life come from the clock of the cell, the clock of the living tissue, the clock of the living individual, and the clock of the life group. Using bionic logic to develop vital signs can only be achieved using structural intelligence. In this way, the hardware equation structure diagram, the software equation structure diagram, the machine society equation structure diagram, and the human-machine society equation structure diagram must all have clocks. To this end, a clock equation system must be established, with a clock embedded inside the central processing unit chip. This includes both the single-machine central processing unit and the central processing unit of the network server. The clock creates a clock equation system, which is in turn used to simulate the life clock. The clock equation variables generated by the clock equation system control a number of different equations and equation unit structure diagrams. The functions of equations and unit structure diagrams changing with time can then be used to express the machine's vital signs. After this, machine reproduction, inheritance, iterative evolution, life cycle, and other life activities can be created according to the machine's vital signs. In this way, the equation unit generator can create genetic information and create a form of heredity. This again illustrates that the structural intelligence machine is a true scientific form of bionic logic, an intelligent machine constructed using the natural laws of life.

2. Research and Development Tools for Structural Intelligence Machines

2.1. Bionic Logic Methodology

2.1.1. Bionic Logic

The steps involved are as follows: ① Finding biological equations. Biological equations are the logical rules of life that have evolved over hundreds of millions of years of life on Earth. They are significantly more ingenious than any artificially created rules. The logical equations for designing logical structures and artificial intelligence must therefore first look for the biological equations of life; ② While looking for biological equations, it is also necessary to search for the relationship between the variables within the equations; ③ Upgrading biological equations and their relationships to structure diagrams composed of logical equations and logical equation relationships; in other words, creating the logical equation structure diagrams.

2.1.2. Simulating Life

A basic bionic logic that forms the primary and intermediate structures of logic, as well as the primary and intermediate structures of artificial intelligence. The steps for simulating life are as follows: ① Finding the biological equations and equation relationships of cells, enabling the formation of a directed diagram file—there will be one directed diagram per topic, with different diagrams indicating the required relationships. This file is the cell's structure diagram, and will include its structure, life activities, and life cycle; ② Simulating the cell's structure diagram to construct the logical equation structure diagram of the primary elements of logical structures and artificial intelligence; ③ Studying the relationships between cells, creating a directed diagram for the relationships between cell structure diagram files to form a primary structure diagram of life, and then simulating this structure diagram to form the primary structures of logical structures and artificial intelligence; ④ Finding the biological equations and relationships for biological tissue, in order to form a directed diagram file—one directed diagram per topic, with different diagrams indicating the required relationships. This file is the structure diagram of the biological tissue, and will include its structure, life activities, and life cycle; ⑤ It will also be necessary to find the biological equations of the relationship between biological tissue and cells. These equations will form a structure diagram that is added to the biological tissue structure file, establishing relationships with other structure diagrams to generate a complete biological tissue structure file; and ⑥ Simulating the structure diagram of the biological tissue to construct logical equation structure diagrams for the intermediate structures of logical structures and artificial intelligence.

The life cycle of advanced logical structures can be simulated as a form of human life, with various life stages. The life cycles of the primary and intermediate structures of advanced logical structures simulate human life, while the life cycle of the brain of advanced logical structures also simulates the human brain cycle. Simulating the clock equation structure diagram file of human beings creates the clock equation structure diagram file for advanced logical structures, allowing the artificial intelligence clock equation structure diagram file to be constructed accordingly.

2.1.3. Simulating Humans

Constructing the advanced structures of logical structures and artificial intelligence requires the simulation of human beings. This is a form of advanced bionic logic, and will consist of the following steps: ① Using the theory of simulating life to simulate human cells and tissues, forming primary and intermediate structures; ② Finding the relationship equations between different tissues, and thus forming a structure diagram connecting different tissue structure files to form an overall intermediate structure; ③ Finding the biological equations and equation relationships of brain tissue, in order to form a directed diagram file—one directed diagram per topic, with different diagrams indicating the required relationships. This file is the structure diagram of the brain tissue, and includes its structure, life activities, and life cycle; ④ It is also necessary to find the biological equations representing the relationship between brain tissue and cells, as well as those between brain tissue and other tissues. These equations form a structure diagram that is added to the brain tissue structure file, establishing relationships with other structure diagrams to generate a complete brain tissue structure file; ⑤ Simulating the structure diagram of brain tissue to construct logical equation structure diagrams for the advanced structures of logical structures and artificial intelligence.

2.1.4. Principles of Bionic Logic

① The simulation of bionic logic can be partially simulated according to actual needs. ② Simulation can be simple or complex depending on the number of iterations employed. ③ It is recommended to simulate as much as is currently known, and to temporarily replace the parts that cannot be simulated with artificial design. Then, as research progresses, these artificial operations can be replaced with simulations, forming a continuous process.

2.1.5. Simulating Society

This theory believes that the environment of logical structures, just like the human environment, is a society. Simulating human society to construct the society of logical structures enables the interaction between people in human society to be simulated, thus constructing the interaction between social logical structures and logical structures, as well as between logical structures and people. Simulation is both iterative and infinite, a constant simulation, and the simulated society of artificial intelligence must be constructed accordingly.

2.1.6. Extended Bionic Logic

In order to create the life-giving logical equation structure diagram of the structural intelligence machine, it is necessary to create logical equation structure diagrams for universal common sense, natural science common sense, and social science common sense. This is also a form of bionics, an extended bionic logic [1].

2.2. Logical Equation Theory

2.2.1. Equation Overview

Logical equations are based on physical, chemical (material equations) and biological equations. These equations or equation sets show the determining effect of certain variables on others. Assigning values to these variables can derive the values of other variables. In physical, chemical and biological equations, these variables describe the structure, life activities, and life cycle of matter or life. When applied to intelligent technology, they describe the structure, life activities, and life cycle of intelligent machines. Logical equations include not only objective equations (such as physical, chemical and biological equations), but also subjective equations, such as belief equations, emotion equations, and others, created by the machine brain. Subjective equations reflect the logic of the human brain and can also become the logic of advanced logical structures and intelligent machine consciousness.

Based on material equations and biological equations, the concept of logical equations is proposed. Logical equations can be divided into three categories: primary-level logical equations use the laws of material equations and cell equations; intermediate-level logical equations use the laws of biological equations of tissues and organs; and advanced-level logical equations use the biological equation laws of the brain.

2.2.2. Definition of Equations

To begin with, all equations within the software and hardware of structural intelligence machines must take the form of functions. However, when solving for values, no distinction is made between functions and variables. Functions and variables are only divided into input assignment variables and output solution variables.

The equations in the knowledge equation structure diagram of the equation unit generator, and the equations in each software diagram, are all individual function programs. These can derive a group of assigned output variables from a group of assigned input variables. The equations in the software diagram can have various required modalities.

2.2.3. Equation Data

1) Class Data

The value set of equation variables in the life-giving logical equation structure diagram stores class data. Class data is defined as follows:

Class Data = Description Set ∧ Attribute Set ∧ Function Set (Formula 1)

The description set is a description of the current situation of the class data. The description set can have a time point in the form of a clock. The attribute set stores the logical equations that determine the attributes, while the function set stores the logical equations that determine the functions. If the attribute set and function set are omitted, leaving only the description set, then the class data becomes raw data.

Class data includes a description set, an attribute set, and a function set; the difference from traditional raw data lies in its class form. Various statistical functions, operations, and processing of class data can be performed on the records of the description, attribute and function fields. Class data can have a simplified form.

2) Data Class Group

All class data in the value set of a variable can be grouped together to form a data class group, also referred to as simply a class group. For example, this group could be a collection of the files of each student in a class, or the registration of each motor vehicle in the city of Guangzhou. A data class group is a new type of database or data warehouse.

3) Mutual Interference of Data Class Groups

Mutual interference of data class groups, abbreviated as mutual interference, is aimed at constructing logical equations. Mutual interference research is an advanced stage of diagram analysis and an important part of empowering equation unit generation to construct logical equations. Mutual interference research requires specific research methods. To begin with, the object sets of both parties involved in mutual interference must first be combined into a single set, which will constitute the future logical equation. Methods are sought based on the laws of this overall set. The social or natural laws of the research object must be combined in the mutual interference research process.

2.2.4. Related Equation Formulas

Standard Form of Logical Equation = Clock Form ∧ Address Form ∧ Communication Form ∧ Modality Form ∧ Equation (Formula 2)

Standard Form of Equation: Function Variable = Formula (Variable 1) ∧ Formula (Variable 2) ∧ Formula (Variable 3) ∧ … ∧ Formula (Variable n) (Formula 3)

Standard Form of Variable = Data Class {(Description Set) ∧ (Attribute Set) ∧ (Function Set)} (Formula 4)

Note that data classes have these forms: (Description Set) ∧ (Attribute Set) ∧ (Function Set), (Description Set) ∧ (Attribute Set), (Description Set) ∧ (Function Set), (Description Set).

Standard Form of Variable Value Set = {Data Class 1 (Description Set) ∧ (Attribute Set) ∧ (Function Set), Data Class 2 (Description Set) ∧ (Attribute Set) ∧ (Function Set), Data Class 3 (Description Set) ∧ (Attribute Set) ∧ (Function Set), …, Data Class n (Description Set) ∧ (Attribute Set) ∧ (Function Set)} (Formula 5) [1]

2.3. The Study of Life-Giving Logical Equation Structure Diagrams

2.3.1. Logical Equation Structure Diagram

This is a directed diagram where a node represents one or more logical equations. If the variable(s) of this logical equation or equation set determine the variable(s) of another logical equation or equation set, then this node will have a directed path to the other node; weights can be added to the paths to indicate their order. Writing out all logical equations and marking all variable relationships with directed paths completes a logical equation structure diagram.

2.3.2. Life-Giving Logical Equation Structure Diagram

Endowing a structure diagram with life by using a clock is the first living diagram theory in human academia, and the first pioneering diagram theory tool of the intelligent era. This structure diagram includes scientific logic, life logic, and subjective logic, and is the main tool for describing intelligent structures and intelligent technology. Intelligent engineering is the process of designing and implementing structure diagrams, while intelligence is represented by life activities and the life cycle of the structure diagram. The research tools, design tools, operating tools, monitoring tools, and modification tools for machine brain structures are all life-giving logical equation structure diagrams. The truth or falsehood of each unit of the logical equation structure diagram is judged using logical mechanics [1] [2].

2.4. Diagram Data Theory

2.4.1. Relevant Definitions

For "Class Data, Data Class Group, Data Class Group Mutual Interference", please see the discussions in Chapter 2.2.3 1) - 3).

2.4.2. Significance of Diagram Data Theory

The diagram data theory uses class data instead of raw data, taking the class data distributed in the variable value sets of each equation structure diagram instead of the raw data concentrated in the database. It uses equation analysis instead of statistical analysis, and an equation unit generator to generate the unit logical equation structure diagram instead of statistical results generated by the database. The associated data is attached to the life-giving logical equation structure diagram. The database is a value set of equation variables, while the associated data analysis consists of logical equation analysis—a type of logical mathematical analysis. The result of this data analysis is not to obtain statistical data, but to empower the equation unit generator to generate a unit equation structure diagram, creating the thinking and new knowledge of intelligent machines.

For example, traditional data analysis often produces indices, such as economic indices. In contrast, diagram data theory uses a life-giving logical equation structure diagram to perform its data analysis. The index calculation formula becomes a structure diagram, turning economic index analysis into economic structure diagram analysis. Complex analysis, clock analysis and equation unit generator analysis are used to generate a unit equation diagram instead of an index, transforming economic index research and extending it to other fields. Structure diagram analysis is used to achieve both comprehensive analysis and cross-domain analysis [3].

2.4.3. Replacing Database Analysis with Equation Structure Diagram Analysis

Traditional data analysis is a concentrated mathematical analysis of raw data within a database, while the data analysis of diagram data is a logical mathematical analysis of class data in the variable value sets of each equation structure diagram. This logical mathematical analysis takes the form of equation analysis, which includes the following key points:

① Structure Diagram Complexity and Analysis: This analysis requires that class data in the variable value set can exist in massive amounts, and that massive class data can exist within the equation structure diagram.

② Clock Equation System and Analysis: This analysis requires data to have life, endowing both equations and data with life, and also that data can be activated or terminated.

③ The equation unit generator creates new equation units to replace traditional data analysis indices. That is, the data analysis of diagram data is completed by the equation unit generator, resulting in the new variable value set and new variables generated by the new unit software diagram, as well as new variables [3].

2.4.4. Inheriting Data Training

Diagram data theory has its own complete set of learning methods, but it still needs to inherit the data training process of information-based intelligent technology. The development of human intelligent technology originated from data training algorithms. It has been proven that data training is still effective for organizing massive amounts of data and constructing logical relationships for machine learning. However, constrained by limited data resources and limited computing power, it is desirable to develop a new method for diagram learning combined with a necessary amount of data training.

2.5. Logical Spacetime Theory

2.5.1. Overview

Logical spacetime theory involves establishing the common sense that matter is actually a logical subject, material phenomena are actually logical phenomena, and the relationship between all matter is actually a logical relationship. Logical phenomena and logical relationships are also logical subjects. To this end, the concept of a logical subject should be developed to also represent time and space. Time, space, and matter can then be represented by a single concept of a logical subject. At the same time, the concepts of logical phenomena and logical relationships can be established; logical spacetime, logic, logical phenomena, logical relationships, and the logical universe can then be used to transform and develop existing mathematical and physical theories. Machine brain structures are required to exist in logical spacetime and the logical universe. The external communication of machine brain structures requires logical networks to transmit logic. Logical networks need to exist in a logical environment, that is, a logical field. The existence of a logical field requires logical spacetime as well as the logical universe to act as the environment for the logical field.

Therefore, the existence or production and use of machine brain structures requires a form of logical spacetime. This logical spacetime needs to be based on the theories of logical mathematics and logical physics. These theories first require the creation of a group of common-sense concepts, such as time, space, and matter being logical subjects, the universe being a logical subject, material phenomena being logical phenomena, the relationships between matter being logical relationships, universal connections being universal logical relationships, and logical phenomena and logical relationships also being logical subjects. Logical equation structure diagrams can then be used to describe everything in the universe, after which bionic logic, logical mathematics, and logical physics can be applied to construct the life-giving logical equation structure diagram of the logical universe. The tools for establishing logical, mathematical and physical theories are the life-giving logical equation structure diagrams and bionic logic.

Machine brain structure engineering is a form of logical engineering, which has two basic requirements: ① The logical engineering product is a logical structure. This structure needs to exist in a logical environment, which in turn needs to exist in logical spacetime; ② The logical engineering process is the process of developing a logical structure, which also needs to exist in logical spacetime.

Bionics refers to using logical equation structure diagrams to simulate all logical subjects, logical phenomena, and logical relationships in the universe.

2.5.2. The Following Are Key Points for the Theory of Logical Spacetime

① The common sense for establishing logical spacetime is that time, space, and matter are all logical subjects, material phenomena are logical phenomena, universal connections between matter are logical relationships, and logical phenomena and logical relationships are also logical subjects.

② Logical subjects are described using three sets: a description set, an attribute set, and a function set, all of which can be attributed to a life-giving logical equation structure diagram. As long as the output variable set contains the union of the three sets, the method for identifying a logical subject is to determine whether the subject can be expressed as a life-giving logical equation structure diagram, and whether the output variable set of this life-giving logical equation structure diagram contains the union of the description set, attribute set, and function set.

③ Three tools are needed to work together when expressing logical subjects using life-giving logical equation structure diagrams, namely bionic logic, life-giving logical equation structure diagrams, and logical mathematics and physics.

Bionic Logic constructs life-giving logical equation structure diagrams through bionics. Bionics refers to finding the structure, activities, and cycle of a logical subject. The life-giving logical equation structure diagram is constructed according to the bionic results of the logical subject. Life here refers to the activity cycle of the logical subject, which can include inanimate logical subjects and life itself. Parts that cannot currently be simulated are designed artificially. The most important work in this process involves the equation unit generator working with the clock equation system to construct the activities and cycles of the logical subject.

Logical mathematics and physics include logical mathematics and logical physics. Logical mathematics is the tool for establishing the equations of the life-giving logical equation structure diagram, while logical physics is the tool for helping bionic logic find the structure of the life-giving logical equation structure diagram.

④ The following logical fields and logical networks are all described using life-giving logical equation structure diagrams; the “life” described here refers to the clock cycle, while the birth and disappearance of logical units are described using the equation unit generator.

2.5.3. The Following Are Key Points for the Theory of Logical Mathematics

① The transformation of traditional mathematics by logical mathematics is reflected in the following two points: one is to change the study of numbers in traditional mathematics to the study of logical equations in logical mathematics; the other is to attribute the research results, research methods, and future research of traditional mathematics to the field of constructing binary and multiple relationships in logical equations.

② The first task of logical mathematics is to express the logical equations of all things in the universe derived from bionic logic. The second task is to study how to improve the ability of the equation unit generator of the life-giving logical equation structure diagram, how to summarize the known logical paths between input and output into logical equation units, and how to create logical equation units to achieve the logical relationship from input to output after knowing the input value and output value. The third task of logical mathematics is to incorporate traditional mathematics into the field of studying logical relationships, and to use traditional mathematics to construct the binary and multiple relationships of logical equation structure diagrams.

2.5.4. The Following Are Key Points for the Theory of Logical Physics

① The transformation of traditional physics by logical physics is reflected in: Transforming physical definitions and theories into logical subjects and creating life-giving logical equation structure diagrams. In the future, the research results of logical physics will be expressed by a combination of logical subjects and their life-giving logical equation structure diagrams.

② There are three tools for the expression and research of logical physics: Bionic logic, life-giving logical equation structure diagrams, and logical mathematics. Bionic logic imitates the spacetime of the universe and all things in spacetime. The life-giving logical equation structure diagram further expresses bionic logic as a structure diagram. The birth, activity, and demise of all things are realized using the equation unit generator. Life is the simulation of the clock cycle of all things. Logical equations are specifically constructed using logical mathematics, while traditional mathematics is used to construct the binary and multiple relationships of equations.

③ Traditional physics, which has become a logical subject, should be used in combination with the logical subjects of newly created logical physics to help bionic logic find the structure of the life-giving logical equation structure diagram.

④ A number of basic concepts need to be created within the realm of logical physics: Time, space, and matter are all logical subjects, material phenomena are logical phenomena, widespread connections are logical relationships, and logical phenomena and logical relationships are also logical subjects. Logical physics can then be used to establish logical spacetime.

⑤ The laws of logical spacetime and logical subjects, logical phenomena, and logical relationship subjects in logical spacetime need to be studied by means of the life-giving logical equation structure diagram of the logical subject theory of logical physics, in order to strengthen the study of bionics. The description set, attribute set, and function set of logical subjects can then be expressed using the life-giving logical equation structure diagram.

In this way, the Logical Structure School uses life-giving logical equation structure diagrams to study and deepen traditional mathematical research, traditional physical research, and traditional logical research, as well as creating and developing the theories of logical spacetime, logical mathematics and physics. In this way, the path of pseudoscience is firmly rejected. However, logical spacetime and logical mathematics and physics are new concepts, and need to be further expanded and developed. They should not be arbitrarily discounted. The development of the intelligent era will inevitably require the two fundamental applied theories of logical spacetime and logical mathematics and physics [3].

2.6. Logical Field Theory

2.6.1. What Is a Logical Field?

A logical field refers to a logical environment. Machine brain structures are required to exist in a logical environment and interact with this environment to mutually enable each other to take shape. The logical environment is also required to exist in logical spacetime and interact with it to mutually shape each other. Logical spacetime is the environment of the logical environment, and both the logical environment and logical spacetime exist in the same spacetime.

A logical field is the environment formed by logical structures, which are themselves logical subjects; interactions exist between logical structures, and various interactions intertwine to form a field. Logical fields include natural fields and social fields, as well as integrated fields that combine both nature and society. The logical field has a logical environmental effect on the logical structure (*i.e.* the logical subject), resulting in a phenomenon of mutual shaping between the logical structure and the logical field. Logical structures interact with the logical field through the perception of logical transmission. Logical transmission is transmitted externally to the logical structure, through a logical network, and internally through a logical bus. The logic perceived by the logical structure is a type of data logic. The logical field can be made into a life-giving logical equation structure diagram. When constructing the structure diagram of a logical field, it is important to note the following key points: ① The required construction tools are bionic logic, life-giving logical equation structure diagrams, and logical mathematics and physics. The role of these three tools is the same as that of the three tools in a general structure diagram; ② In the first instance, it is necessary to create a life-giving logical equation structure diagram for the logical subject in the logical field. The determination of the logical subject to be selected will depend on the requirements of the research object; ③ The transmission direction of the equation units of the logical subject within the logical field can then be marked with a directed path. After marking, the structure diagrams of each logical subject form a whole diagram, comprising the structure diagram of the logical field; ④ The clock equation system and equation unit generator of the life-giving logical equation structure diagram are distributed within the structure diagrams of each logical subject, and within the structure diagram of the logical field; ⑤ If the research object is one or several logical subjects in the logical field, the mutual shaping of the research object and the logical field is the directed path of the exchange of equation units between the research object and other logical subjects. In other words, the research object and the logical field will share a structure diagram; ⑥ The equation unit here refers to constants, variables, equations, equation sets, and structure diagrams of different scales. Each logical subject can transmit various equation units to other subjects, while the directed transmission path can be marked with weights to indicate their order.

2.6.2. Modified Transmission

After an equation unit is transmitted from one logical subject to another, the equation unit issued by the logical subject receiving the equation unit will modify and transmit the equation unit sent to itself by the previous logical subject. The process of modifying and transmitting is as follows: a logical subject transmits an equation unit to a second logical subject, thus changing the structure diagram of the second logical subject. The second logical subject then changes the equation unit transmitted to the third logical subject due to the change in its own structure diagram. The change from the second to the third is due to the change from the first to the second. What the second transmits to the third is the modified transmission of what the first transmitted to the second [3].

2.7. Logical Network Theory

2.7.1. What Is a Logical Network?

① A logical network is an attribute of a logical field. A logical field is composed of logical structures and the interactions between them. This creates a network composed of logical propagation and logical propagation phenomena, which is called a logical network.

② A logical network propagates logic. Many networks mentioned earlier are logical networks: The propagation of radio waves, the interaction of microscopic particles, the water network on the Earth's surface, the propagation of forces inside the Earth's crust, the human highway network, and the human legal network are all examples of logical networks. The radio waves, microscopic interactions, water energy, crustal kinetic energy, goods transported by highways, and legal constraints that are propagated are all forms of logic. In this sense, a logical network propagates logic.

Note that the manifestation of logic is a logical subject; that is to say, the inter-transmission between logical subjects is also the logic of a logical subject.

③ The carrier for propagating logic is a logical subject, and includes constants, variables, equations, and structure diagrams of various sizes, that is, equation units. The logical subject is the form of propagation, which can be a truck, a court, or even a postal package. However, note that logic is the essence of propagation, a logic that has an impact on the logical subject receiving the propagation, while the impact is on the equation structure diagram of the logical subject.

④ Logical propagation is a form of modified transmission. The process is as follows: a logical subject receives logical propagation, and that logic affects and changes the equation structure diagram of the logical subject. The changed logical subject outputs its own logic to the new logical subject according to its own life activity laws.

2.7.2. The Role of Logical Networks

① Logical networks are an important attribute of logical fields. In order to create a logical field, it is necessary to first build a logical network. The quality of a logical network reflects the quality of the logical field, while the function of a logical network is to transmit logic. This transmission is in the form of a logical subject and includes various equation units.

② To fully develop the intelligent era, it is necessary to study and build logical networks. This is not as simple as simply having the Internet and 5G. The logical networks that humans face and build, both natural and artificial, are equally needed by intelligent machines. For example, intelligent machines must also face the gravitational network of the universe, and the supervision of intelligent machine societies must also rely on a legal network.

2.7.3. How Can We Create and Use Logical Networks?

Both the development of humans and the development of intelligence require the construction and use of logical networks. The nature of logical networks to propagate logic means that their creation and use include all aspects of human engineering, including aerospace engineering and deep-sea engineering, legal engineering and educational engineering, as well as market engineering for intelligent development and legal engineering for intelligent supervision.

2.7.4. Theory

① The environment of a logical network is a logical field, while the environment of a logical field is logical spacetime. The form of expression and required research tools for logical networks are bionic logic, life-giving logical equation structure diagrams, and mathematical and physical logic. Logical physics works together with bionic logic to create life-giving logical equation structure diagrams, while logical mathematics is used to construct equations and equation unit generators.

② A logical network is a weighted directed path of equation units in a life-giving logical equation structure diagram, with the weighting indicating the order. Equation units include constants, variables, equations, and equation structure diagrams of different scales. A logical network is a structure diagram composed of weighted directed paths and nodes, with the weighting indicating their order. For a logical subject, its external network is called the extranet network, with the nodes of the extranet network consisting of individual logical subjects. Its internal network is called the intranet network, with its nodes consisting of individual equations or equation units of different sizes.

③ The transmission path of a logical network, including the weighted directed path and its nodes, is known as a logical channel, with the weighting indicating the order. Logical channels include both natural channels and artificial channels. The former transmits logical subjects, for example, sunlight from the sun falling on the Earth, or a train transporting passengers from one city to another, while the latter currently transmits equation units, such as the Internet or the signal lines inside the computer. The intranet network of a machine brain structure needs to simulate human nerves. The advanced development stage of artificial channels also needs to transmit logical subjects [3].

3. A Novel Central Processing Unit

3.1. Innovative Central Processing Units—Full Diagram of Processor Equation Structure

Full Diagram of Processor Equation Structure = Knowledge Equation Structure Diagram + Life Clock Unit Diagram and Subsidiary Diagrams + Address Unit Diagram and Subsidiary Diagrams + Communication Unit Diagram and Subsidiary Diagrams + Modality Control Unit Diagram and Subsidiary Diagrams (Formula 6)

In the operating process of the full diagram, first, the equation unit generator controls the clock and other components of the processor. In other words, the equation unit generator controls the entire intelligent machine. The equation unit generator is used to realize the two functions of diagram data and diagram learning. Then, the life clock unit diagram controlled by the equation unit generator endows the machine with life and vital signs. After this, the knowledge equation structure diagram endows the equation structure diagram of the software diagram. In combination with their various subsidiary diagrams, the life clock unit diagram, the address unit diagram, the communication unit diagram, and the modality control unit diagram endow the software diagram with various clock forms, address forms, communication forms, and modality forms to realize the functions of diagram clock, diagram address, diagram communication, and diagram modality. Finally, the software diagram is realized, and various forms of diagram thinking, including diagram analysis, diagram reasoning, and diagram control, can be realized.

Note that the novel central processing unit should also feature the various required registers; the management of these registers will still use clock forms, address forms, communication forms, modality forms, and different types of equations.

3.2. Equation Unit Generator

For structural intelligence machines, the equation unit generator (referred to as the generator) takes up a key position—it acts as controller, the main body of machine learning, marking the level of structural intelligence.

3.2.1. Knowledge Equation Structure Diagram

Standard Diagram of Knowledge Equation Structure Diagram = Common Sense Structure Diagram + Diagram Obtained from Data Training + Software Diagram New Knowledge + Generator Logical Learning (Formula 7)

3.2.2. Connection with Software Diagrams

The main function of the equation unit generator is to generate unit software diagrams and to control the clock and other components of the central processing unit by transmitting and receiving information, thereby controlling the entire structural intelligence machine.

3.2.3. Algorithm for Equation Unit Generator

① Algorithm for generating unit software diagrams: To generate a unit software diagram, the generator first sends notifications to the life clock unit diagram, address unit diagram, communication unit diagram, and modality unit diagram to obtain the clock form, address form, communication form, and modality form, respectively. After this, the equations of the knowledge equation structure diagram form a standard logical equation to generate the software diagram.

② Machine metabolism: The life clock unit diagram and its subsidiary diagrams controlled by the generator can make the clock form return to zero, which is the equivalent of deleting the logical equation or unit software diagram where the clock form is located. This is an important condition for the real-time dynamic life of the software diagram. The behavior of the generator as it creates unit software diagrams for software diagrams or deletes unit software diagrams by controlling the clock is called machine metabolism.

③ Common sense learning algorithm: The common sense equation structure diagram includes: logical spacetime, logical mathematics, logical physics, natural science common sense, and social science common sense. This requires the artificial construction of existing knowledge into equation structure diagrams, which in turn requires a huge amount of work, possibly the strength of an entire country. This will be used to install common sense equation structure diagrams on network servers for network sharing, similar to a library.

④ Data training algorithm: Intelligent technology started with data training algorithms, a form of information-based technology that is not entirely useless. It is often necessary to remove the dross in order to extract the essence. Structural technology retains data training technology, performing feasible and necessary data training on massive amounts of data, and supplementing the common sense learning algorithm.

⑤ Software diagram new knowledge algorithm: Software diagrams generate new diagrams that the generator does not yet have by means of diagram thinking, that is, analysis and reasoning; these new diagrams are added to the generator structure diagram.

⑥ Generator logical learning algorithm (the core technology of structural intelligence): This algorithm combines artificially designed algorithms with the software diagram new knowledge algorithms to develop the ability of intelligent machines to turn various modalities of information into logic. It then stores this logic in the form of logical diagrams in the standard diagram of the knowledge equation structure diagram, thus creating logical learning for the generator. This is a process of continuous iteration.

⑦ Machine heredity created by the generator: The generator can create a real-time dynamic machine life environment by controlling the clock. In this environment, the generator can autonomously and automatically generate two executable directories using artificial algorithms. One is the directory of the knowledge equation structure diagram, while the other is the directory of software and hardware control functions; both of these can be used as machine genes. When machine reproduction technology is developed in the future, machine genes can be passed on to the next generation of intelligent machines, producing machine heredity.

3.2.4. Diagram Learning Theory

1) Main Body of Learning: The Equation Unit Generator

The main body of diagram learning is the equation unit generator within the new central processing unit, which is itself divided into three types of equation unit generators:

① Equation Unit Generator for the Intelligent Machine Brain

This equation unit generator simulates the learning of the human brain, producing the knowledge and reactions of the human brain. It is a type of learning for knowledge and reactions on how to perform diagram thinking.

② Equation Unit Generator for the Intelligent Robot

This equation unit generator simulates the learning of the human body, producing the associated knowledge and reactions. The knowledge and reactions of the human body are the control-oriented knowledge and reactions of diagram thinking. That is to say, the robot uses the equation unit generator to generate new equation structure reaction diagrams and software diagrams to control various intelligent devices connected to the machine brain, including information reasoning learning (diagram reasoning) and structural reasoning learning (diagram control).

③ Equation Unit Generator Group for the Intelligent Society

This is a group of equation unit generators that simulates human society, with a leadership group and a subordinate group. The leadership group learns information reasoning (diagram reasoning) and social reasoning (diagram control) in order to lead society, while the subordinate group bears the information reasoning and social reasoning of the leadership group. The information reasoning of the leadership group often refers to the generation of policies, using the law to implement social reasoning. The information reasoning learning of the leadership group machine involves the creation of machine policies and machine laws. The social reasoning learning involves ways of promoting machine policies and machine laws to the subordinate group.

2) Form of Learning: Equation Structure Reaction Diagram

Traditional machine learning uses massive data and enormous computing power to devise program equations, which are in turn used to provide various forms of required information. This technology has been criticized for having three unscientific aspects. The first and second aspects are that the cost required for massive data and enormous computing power is not economically viable. The third is that its intelligent development path is different from that formed by the universe. These issues suggest the future is bleak for traditional machine learning. The intelligent development path designed by the universe is the path of humankind, while machine learning is more suited for training lower animals. Limited by traditional hardware conditions, the path of traditional machine learning is certainly unsuitable, and data training will be unable to produce advanced intelligence.

Diagram learning proposes the theory of generator learning and generator social learning based on the needs of the intelligent era, and looking forward to the further development of hardware. Here, the theory of generator learning is studied. This is a learning theory of robots. The robot's brain tissue (equation unit generator) receives external information, performs information reasoning through learning, and then forms learning results in the form of an equation structure reaction diagram. The equation structure reaction diagram is used to create a software diagram, while the software diagram output is used to perform structural reasoning to control the robot's intelligent machinery.

The equation structure reaction diagram, that is, the new knowledge equation structure diagram, is the both the process and result of the learning proposed by diagram learning theory. The learning process involves information reasoning. A particular strategy is used to perform information reasoning on external information, with an equation structure reaction diagram formed through information reasoning. This diagram is the knowledge learned by the generator. The equation structure reaction diagram is still a life-giving logical equation structure diagram, whose role is to change the generator's original knowledge equation structure diagram. The software diagram generated by the new knowledge equation structure diagram is used to perform structural reasoning, that is, the software diagram's thinking output is used to control the robot's structural work. The new equation structure reaction diagram is then used to change the generator's original knowledge equation structure diagram. The new knowledge equation structure diagram is not only able to create the existing input to output of the software diagram, but can also generate new variables. The output of the new software diagram is then used to control intelligent machinery to realize new structural reasoning. This simulates the process of human thought and the generation of new knowledge.

3) Learning Strategies

Using external information to construct an equation structure reaction diagram is what constitutes the learning strategy of diagram learning, and includes the following.

1. Bionic Logic Strategy

For the bionic logic strategy, the first step is the bionics of the human body. Scientific research, such as biology, medicine, and psychology, is used to study the laws devised by humans, the laws of how humans think about problems, and the laws of how the human brain controls the body, including the laws of how human senses work. The second step is to turn these various laws into logical equations, and then combine the logical equations to construct an equation structure reaction diagram. The equation structure reaction diagram can then be stored in the knowledge equation structure diagram of the equation unit generator to be called up when needed. This is the information reasoning of bionic logic, a process also referred to as learning. The third step is to call on the equation structure reaction diagram to generate a software diagram. Using the software diagram's diagram thinking output to control the robot's structure work constitutes the process of structural reasoning.

2. Logical Physics Strategy

For the logical physics strategy, the first step is to study different scientific laws of the universe. The second step is to turn these laws into logical equations, and then combine these equations to construct an equation structure reaction diagram. This diagram is then stored in the knowledge equation structure diagram of the equation unit generator to be called upon when needed. This is the information reasoning of logical physics, a process also referred to as learning. The third step is to call upon the equation structure reaction diagram to generate a software diagram. Using the software diagram's thinking output to control the robot's structure work is a form of structural reasoning.

3. Logical Mathematics Strategy

The bionic logic strategy and the logical physics strategy are both forms of external learning, though the logical mathematics strategy is also an autonomous learning strategy. The following autonomous learning methods can be created using the laws of logical mathematics:

① The process of generating a new unit software diagram involves determining a batch of input variables, then finding the required output variables and the path from the input variable set to the output variable set. In this way, the logical relationship between the input variable set and the output variable set can be obtained, and a new unit software diagram, that is, a new equation structure reaction diagram, can be automatically generated based on this logical relationship. The new equation structure reaction diagram is used to change the generator's original knowledge equation structure diagram. This new diagram can not only realize the existing input to output, but can also generate new variables. The output of the new software diagram is then used to control intelligent machinery to create new structural reasoning. This simulates human thought and the generation of new knowledge resulting from it.

② The process of generating a new equation unit can also determine a batch of input variables, then find the values of the required output variables and the path that achieves logical reasoning between the two values. In this way, the logical relationship between the input variable set and the output variable set can be obtained, and a new equation structure reaction diagram can be automatically generated based on this logical relationship. The new diagram is used to change the generator's original knowledge equation structure diagram, thus not only realizing the existing input to output, but also generating new variables. The output of the new software diagram is used to control intelligent machinery to create new structural reasoning. This is also a simulation of human thought and the generation of new knowledge from human thought.

③ The equation unit generator can also work with the communication system.

A new equation structure reaction diagram is generated by the interaction of newly input equation units and existing equation units, or a new output value generated by newly input variables generates a new equation structure reaction diagram, creating new autonomous learning methods.

④ Advanced equation unit generators can even create new equation structure reaction diagrams based on self-defined input sets and output sets.

4. Logical Data Analysis Strategy

The strategy of logical data analysis is also an autonomous learning strategy. Logical data analysis has similarities with the current method of large model data training, since both use data to generate program equations, but there are also differences: ① The path of large model data training involves training, forming equations, and applying equations, while logical data analysis can form equations immediately; ② Large models require equations to be input first before applying them, indicating an external learning strategy, while logical data analysis forms equations immediately and also applies them immediately, indicating an autonomous learning strategy; ③ The equation application of large models relies on flowchart programs, while the equation application of logical data analysis relies on new equation structure reaction diagrams. There is a difference between flowcharts and structure diagrams in these two programs; ④ Large models require huge amounts of data and computing power, and develop intelligence using the same methods as those to train lower animals, an unsuitable way of developing intelligence in an industrial society. Logical data analysis uses much less data and computing power than large models. Logical data analysis uses the learning methods of biomimetic humans, while the learning and growth methods of humans who first learn knowledge and then form abilities through thinking represent the correct path for the development of autonomous learning for intelligent machines in an intelligent society.

There are several strategies for logical data analysis:

One type of analysis is the known equation structure diagram, where the value set of one or more unknown variables is obtained from the value set of one or more known variables.

Another type of analysis is the unknown equation structure diagram. Several class groups are set as variable value sets, and the interrelationship of these class groups, that is, their mutual interference, is studied. The research results empower the equation unit generator to construct new knowledge equation structure diagrams and structure reaction diagrams [3].

3.2.5. Proposed Mutual Interference Algorithms

These mutual interference algorithms all utilize an important phenomenon, that is, the data of each class group appearing in an orderly manner according to time points; they also use the time point as a clock for constructing algorithms.

1) Mutual Interference Algorithm 1

Scenario: This is the simplest form of mutual interference, where the respective data of two class groups appear in an orderly manner according to time points.

① Finding the different class groups of the mutually interfering data classes. The data of each class group changes according to the time point. The data of different class groups is divided into different groups according to the time point, while the data within each group is the respective data for the same time point from different class groups.

② The groups are numbered according to the order of the time points. The first group is programmed with equations; all logical equations are listed as part of this process, which may involve a huge number of equations. Then, the logical relationships between the data in the second group are used to filter the logical equations in the previous group. By analogy, the logical relationships between the data in the subsequent group are used to filter the logical equations generated by the previous group. The final logical equations obtained are the logical equations or unit equation diagrams of mutual interference. If a logical equation is not obtained at the end, then the mutual interference must have been made in error.

2) Mutual Interference Algorithm 2

Example Scenario: A mutual interference study of the epidemic situation in the cities of Hong Kong and Shenzhen. Since the nature of the two data sets is the same, a simple set merging is possible, and a mutual interference algorithm can be generated.

① Find the two class groups of Hong Kong and Shenzhen that exhibit mutual interference among the data class groups. The data of each class group changes according to the time point. Obviously, at the same time point, there will be one data point for the Hong Kong class group and one for the Shenzhen class group. These two data points will be formed into one group.

② The groups are numbered according to the order of the time points. First, the Hong Kong data of the first group is programmed using equations, with all logical equations listed. This may constitute a huge amount of data. Then, the logical relationship of the Shenzhen data from the first group is used to filter the logical equations generated by the Hong Kong data. After that, the logical relationship of the Hong Kong data from the second group is used to filter the logical equations generated by the first group, after which the Shenzhen data from the second group is used to filter the various logical equations generated by the Hong Kong data in the second group. By analogy, the logical relationships between the data in the latter group are used to filter the logical equations generated by the previous group. The final "Equation (Hong Kong Data) ∧ Equation (Shenzhen Data)" for each logical equation obtained is the mutual interference logical equation or unit equation diagram. If a logical equation is not obtained at the end, then the mutual interference will have been in error.

3) Mutual Interference Algorithm 3

Scenario: A mutual interference study of a water quality data class set and an aquatic product data class set in a fishpond requires that the water quality data class set be added to each element of the aquatic product data class set. This situation can generate a mutual interference algorithm.

① The grouping method for this scenario is as follows: each group has the form "{Water Quality Data Class Set} + One Data Point from the Aquatic Product Data Class Set}".

② The groups are numbered as required. The first group is programmed with equations, after which all logical equations are listed, which may represent a huge amount of data. Then, the logical relationships between the data in the second group are used to filter the logical equations in the first group. By analogy, the logical relationships between the data in each subsequent group are used to filter the logical equations generated by the previous group. The final logical equations obtained are thus the logical equations or unit equation diagrams of mutual interference. If a logical equation is not obtained at the end, then the mutual interference will have been in error.

4) Mutual Interference Algorithm 4

Scenario: There are several data class groups: housing price data class group, vegetable basket data class group, employment data class group, income data class group, stock price data class group, and people's level of satisfaction data class group; the mutual interference laws between these groups are unknown.

① The housing price data class group data, vegetable basket data class group data, employment data class group data, income data class group data, stock price data class group data, and people's level of satisfaction data class group data at the same time point form a group.

② The groups are numbered according to the order of the time points. The first group is first programmed with equations, with all logical equations listed; this may represent a huge amount of data. Then, the logical relationships between the data in the second group are used to filter the logical equations in the first group. By analogy, the logical relationships between the data in subsequent groups are used to filter the logical equations generated by the previous groups. The final logical equations obtained are the logical equations or unit equation diagrams of mutual interference. If a logical equation is not obtained at the end, then the mutual interference will have been in error.

5) Mutual Interference Algorithm 5

Scenario: Establishing the mutual interference index of various data class groups affecting atmospheric temperature.

① The indices of various data class groups that affect atmospheric temperature at the same time point form a group.

② The groups are numbered according to the order of the time points. The first group is programmed with equations, with all logical equations listed; this may represent a huge amount of data. Then, the logical relationships between the data in the second group are used to filter the logical equations in the previous group. By analogy, the logical relationships between the data in the subsequent group are used to filter the logical equations generated by the previous group. The final logical equations obtained are thus the logical equations or unit equation diagrams of mutual interference. If a logical equation is not obtained at the end, then the mutual interference will have been in error.

3.2.6. Mutual Interference of Data Class Groups Empowers the Equation Unit Generator to Construct Unit Software Diagrams

Mutual interference research can empower the equation unit generator to construct unit software diagrams, creating its main logical learning algorithm (the core technology of structural intelligence), and introducing various advanced mathematical tools. There are at least two situations for the mutual interference operation of data class groups. In one situation, an operation is designed to connect each data class group into an overall set. The dynamic changes of the overall set of data class groups caused by changes in time points are used to find the mutual interference laws, which are expressed as logical equations. In other situations, the mutual interference laws are known. According to these laws, a connection operation is designed to connect each data class group into an overall set. This set can be expressed as a logical equation, with each data class group corresponding to a variable [3].

3.3. Clock

3.3.1. Clock Equation Structure Diagram

Standard Diagram of Clock Equation Structure Diagram = Life Clock Unit Diagram + Address Clock Unit Diagram + Communication Clock Unit Diagram + Generator Clock Unit Diagram + Modality Clock Unit Diagram + Unit Software Diagram Clock (Formula 8)

Clock Form = Time Quantity + Access Quantity (Formula 9)

Time Quantity = Address Clock + Communication Clock + Generator Clock + Modality Clock + Unit Software Diagram Clock (Formula 10)

3.3.2. Connection with Software Diagrams

The standard diagram of the clock equation structure diagram generates clock forms and transmits them to the standard form of logical equations and unit software diagrams, thus controlling the logical equations and unit software diagrams where the clock forms are located. The access quantity is determined by the variable of the logical equations and unit software diagrams, where the clock forms are located when accessing the clock. When a logical equation and unit software diagram terminates, the access quantity reports to the life clock unit diagram and the generator clock unit diagram.

3.3.3. Clock Algorithm

The clock algorithm consists of the following steps:

① The life clock unit diagram starts working, giving the intelligent machine life and creating its vital signs.

② The equation unit generator issues a command to generate an equation unit to the life clock unit diagram, after which the life clock unit diagram creates the address clock unit diagram, communication clock unit diagram, generator clock unit diagram, modality clock unit diagram, and unit software diagram clock according to the command. Then, the unit software diagram clock issues a clock form to the corresponding logical equation and unit software diagram. If this logical equation and unit software diagram requires address clock variables, communication clock variables, generator clock variables, or modality clock variables, it accesses the corresponding diagram through the access quantity of the clock form to obtain the clock value. If this logical equation or unit software diagram terminates, then the access quantity of its clock form will output a reply to the life clock unit diagram. The life clock unit diagram then informs the equation unit generator of the termination through the generator clock unit diagram.

③ The clock algorithm needs to be designed and implemented artificially.

3.3.4. Diagram Clock Theory

For the new central processing unit, the clock is controlled by the equation unit generator. The clock then also controls the generated software diagrams with clock forms. Its operating principle is that of the diagram clock theory. In this theory, the working object is the clock equation structure diagram synthesized from the main diagram, life clock unit diagram and the subordinate diagram address clock unit diagram, communication clock unit diagram, generator clock unit diagram, modality clock unit diagram, and unit software diagram clock. To start with, the main life clock unit diagram must create a life clock, which in turn creates machine life and vital signs. The life clock is also an important condition for realizing the machine life cycle and the generator's heredity, because the clock can stipulate the life cycle, and mark the time for the generator's knowledge structure diagram directory and software and hardware control function directory. After this, the main diagram, the life clock unit diagram, is triggered by the equation unit generator to generate each subordinate diagram, and works with the subordinate diagrams to issue various clock forms. At the same time, it also accepts the access quantity and replies, thereby creating a real-time dynamic software diagram environment. Since it is an iterative process, the diagram clock theory needs to be implemented through an artificially designed algorithm.

3.3.5. Life Cycle

① Intelligent Machine Clocks

Intelligent machines have three levels of clocks: the primary element level clock, the intermediate level clock in the tissues and organs, and the advanced level clock in the brain. Each clock determines the life cycle of its corresponding parts.

② Life Cycle of the Machine Brain Structure

Taking humans as an example, intelligence exists when the brain is alive and is lost when it is dead. The core of the advanced structure of intelligent machines is the machine brain structure. The machine brain structure forms the brain, and this brain has a clock. The life cycle variable of this clock is determined by a clock form. The brain structure is responsible for using the clock to calculate the life cycle variable, and in turn, the life cycle variable determines whether the brain can survive and whether advanced intelligent machines have intelligence. The clock form of the machine brain structure possesses both information reasoning (diagram reasoning) and structural reasoning (diagram control) capabilities [3].

③ Implementation Method of the Life Cycle

The structural intelligence machine reflects the diagram control of the machine by the software diagrams of the brain structure, intermediate structure, and primary structure. The life cycle of the machine needs to be reflected in the life cycle of the software diagram, while the life cycle of the software diagram is formulated by the generator based on its own knowledge structure diagram. Then, the generator transmits information to the clock, and the clock form generated by the clock reflects and realizes the life cycle.

3.4. Address Manager

3.4.1. Address Equation Structure Diagram

Standard Diagram of Address Equation Structure Diagram = Address Unit Diagram + Generator Address Unit Diagram + Address Unit Software Diagram (Formula 11)

Address Form = Address Quantity + Access Quantity (Formula 12)

Address Quantity = Storage Address (Formula 13)

3.4.2. Connection with Software Diagrams

To begin with, the generator issues a command to the address unit diagram, and the address unit diagram creates the address form. Then, this address form is added to the standard form of the logical equations and unit software diagrams through the address unit software diagram to allocate storage space. When new storage space is needed, the access quantity accesses the address unit software diagram and transfers it to the address unit diagram in order to increase storage space. When an equation or unit software diagram terminates, the access quantity still informs the address unit software diagram and the generator address unit diagram. The address unit software diagram then tells the address unit diagram to reclaim the storage space.

3.4.3. Address Manager Algorithm

This is the technology required to enable the address unit diagram to efficiently manage storage space.

3.4.4. Diagram Address Theory

The operating principle of the address manager within the new central processing unit is that of the diagram address theory. Objects include the address equation structure diagram, synthesized from the main diagram, address unit diagram and the subordinate diagram generator address unit diagram, and the address unit software diagram. The address unit diagram is triggered by the generator and combines with the subordinate diagrams to generate various address forms to allocate or reclaim storage space. It also works with the subordinate diagrams to reply to the access quantity and manage storage space, realizing both storage management and space allocation. The diagram addresses theory that needs to be implemented through an artificially designed algorithm, and is an iterative evolution process. Existing storage management algorithms can be used for reference.

3.5. Communication Controller

3.5.1. Communication Equation Structure Diagram

Standard Diagram of Communication Equation Structure Diagram = Communication Unit Diagram + Generator Communication Unit Diagram + Communication Unit Software Diagram (Formula 14)

Communication Form = Post Office Function + Access Quantity (Formula 15)

3.5.2. Connection with Software Diagrams

The generator starts by issuing a command to the communication unit diagram, which in turn creates the communication form. This communication form is then added to the standard form of the logical equations and unit software diagrams through the communication unit software diagram, creating a type of post office. When new post office functions are needed, the access quantity accesses the communication unit software diagram and transfers it to the communication unit diagram to increase the post office. When an equation or unit software diagram terminates, the access quantity still informs the communication unit software diagram and the generator communication unit diagram. The communication unit software diagram then tells the communication unit diagram to delete the post office and complete the subsequent operations.

3.5.3. Communication Controller Algorithm

This involves studying the communication unit diagram's ability to establish post offices and post office functions, making it both small and efficient. The clock and storage space required by the post office are obtained through the clock form access quantity and address form access quantity of the logical equation and the unit software diagram, where the post office is located, respectively.

3.5.4. Diagram Communication Theory

The operating principle of the communication controller in the new central processing unit is that of diagram communication theory. Objects include the communication equation structure diagram, synthesized from the main diagram, communication unit diagram, subordinate diagram generator communication unit diagram, and the communication unit software diagram. The communication unit diagram is triggered by the generator and works with the subordinate diagrams to generate various communication forms and create post office functions. When new post offices and post office functions are needed, the access quantity accesses the communication unit software diagram and transfers it to the communication unit diagram to increase the number of post offices. It also communicates with the subordinate diagrams to reply to the access quantity and manage the post offices, their functions, and their storage methods. The diagram communication theory needs to be implemented through an artificially designed algorithm, and is also an iterative process. Existing Internet communication algorithms can be used for reference.

Communication takes the logical field and logical network as the logical environment. The logical field provides the nodes of the extranet communication structure diagram, while the logical network provides its directed paths. Both the intranet communication structure diagram and the extranet communication structure diagram are directed diagrams. The life-giving logical equation structure diagram is a tool for describing and activating not only logical fields, logical networks and logical subjects, but also the life communication process itself. Post offices are nodes of the intranet communication structure diagram, a post office where a logical subject sends equation units to the extranet, and also a node within the logical network communication structure diagram.

3.6. Modality Expresser

3.6.1. Modality Equation Structure Diagram

Standard Diagram of Modality Equation Structure Diagram = Modality Control Unit Diagram + Text Modality Implementation Diagram + Image Modality Implementation Diagram + Speech Modality Implementation Diagram + Video Modality Implementation Diagram + Spatial Modality Implementation Diagram + Other Modality Implementation Diagrams (Formula 16)

3.6.2. Connection with Software Diagrams

The equation unit generator first sends a command to the modality control unit diagram, after which the control unit diagram generates a modality form and adds it to the standard form of the logical equations. When the description set, equation, or unit software diagram of a variable requires a modality expression, a request is sent through the access quantity of the modality form to the corresponding modality implementation diagram of the standard diagram of the modality equation structure diagram, after which the modality implementation diagram outputs the modality based on the description set and its own function.

3.6.3. Modality Expresser Algorithm

The needs of intelligent technology for multi-modality expression and multi-modality thinking can only be satisfied by making use of the true essence of existing technology. This will enable the development of text modality implementation diagrams, image modality implementation diagrams, speech modality implementation diagrams, video modality implementation diagrams, spatial modality implementation diagrams, and other modality implementation diagrams by means of artificial design and the generator's new intelligent knowledge.

3.6.4. Diagram Modality Theory

① Modality Control Unit Diagram

This is the main diagram of the modality expresser, which generates the modality form of logical equations or unit software diagrams. In the real-time dynamic operation of the software diagram, its modality form is able to send the access quantity to the modality control unit diagram. This diagram analyzes the access quantity and leads the path to the corresponding subordinate working diagram based on the analyzed information, providing the required modality expression needed by the unit software diagram.

② Subordinate Diagrams of the Modality Control Unit Diagram

The text modality implementation diagram, image modality implementation diagram, speech modality implementation diagram, video modality implementation diagram, spatial modality implementation diagram, and other modality implementation diagrams are all subordinate to the modality control unit diagram. These diagrams also have various thought paths, with each representing a specific method of modality expression, and various equations of the path being different processes of this method. These subordinate diagrams are artificially designed and need to be continuously iterated.

③ Application of Modality

The role of the modality expresser is to allow the unit software diagram to be expressed as a certain modality as needed, which is in line with the concept that human thought objects all feature modalities. It should be noted that the logical equations or unit software diagrams acted upon by the modality expresser all have clock forms, meaning the expressed modality will be affected by the clock. It is also a real-time dynamic form of modality, thus expressing the same characteristics as human thought.

4. Studying the Software Logical Equation Structure Diagram

4.1. Diagram Thinking Theory

Diagram thinking theory involves studying the life-giving software logical equation structure diagram (referred to as a software diagram). Diagram thinking includes three steps: diagram analysis, diagram reasoning, and diagram control.

Diagram thinking is based on the following two formulas:

① Forward Thinking: Assigned Input Variables + Logical Relationship between Input Variables and Output Variables = Solved Output Variables (Formula 17)

② Reverse Thinking: Assigned Output Variables + Inverse Relationship of the Logical Relationship between Input Variables and Output Variables = Solved Input Variables (Formula 18)

4.1.1. Diagram Analysis Theory

Diagram analysis is the first step in diagram thinking. The purpose of diagram analysis is to find the logical relationship or the inverse relationship of the logical relationship between input variables and output variables in a real-time dynamic software diagram environment, while the difference between various forms of diagram thinking lies in the difference in diagram analysis. The following will discuss the various types of diagram thinking.

4.1.2. Diagram Reasoning Theory

Diagram reasoning is the second step in diagram thinking. The processes of all forward and reverse diagram reasoning are the same, and are all based on the logical relationship or the inverse relationship of the logical relationship between input variables and output variables found by diagram analysis. The output variables or input variables can be solved according to the assigned input or output variables.

4.1.3. Diagram Control Theory

Diagram control is the third step in diagram thinking. Based on the output or input variables solved by diagram reasoning, the study focuses on the effect of the solved variables on the controlled object.

4.2. Equation Diagram Analysis

Equation diagram analysis refers to the logical relationship between input and output variables, which is the relationship represented by a logical equation or a logical equation set, and involves finding the logic or reverse logic of this relationship.

4.2.1. Forward Analysis

Forward equation diagram analysis involves studying the logical relationship represented by a logical equation or a logical equation set.

4.2.2. Reverse Analysis

Reverse equation diagram analysis involves studying the inverse relationship of the logical relationship represented by a logical equation or a logical equation set.

4.3. Path Diagram Analysis

Path diagram analysis refers to the logical relationship between input and output variables forming a path within a real-time dynamic software diagram. This features a starting equation, several intermediate equations, and an ending equation.

1) Forward Analysis

Forward path diagram analysis involves studying the logical relationship represented by the equation path.

2) Reverse Analysis

Reverse path diagram analysis involves studying the inverse of the logical relationship represented by the equation path.

4.4. Parallel Diagram Analysis

Parallel diagram analysis involves studying the logical relationship between input variables and output variables, as well as the multiple parallel paths they form in a real-time dynamic software diagram. Each path features a starting equation, several intermediate equations, and an ending equation.

4.4.1. Forward Analysis

Forward parallel diagram analysis involves studying the logical relationships represented by each parallel equation path, then performing forward diagram reasoning to solve the output variables of each parallel equation path based on the assigned input variables and the logical relationship between the input and output variables represented by each parallel equation path obtained by means of diagram analysis. Forward diagram control can then be performed to study the effect of the output variables solved by each equation path on the controlled object.

4.4.2. Reverse Analysis

Reverse parallel diagram analysis involves studying the inverse of the logical relationships represented by each parallel equation path, then performing reverse diagram reasoning and solving the input variables based on the assigned output variables of each parallel equation path and the inverse of the logical relationship between the input and output variables represented by each parallel equation path obtained from diagram analysis. Reverse diagram control is then performed to study the effect of the input variables solved by each parallel equation path on the controlled object.

4.5. Cross Diagram Analysis

Cross diagram analysis refers to the logical relationship between input and output variables, creating multiple parallel paths in a real-time dynamic software diagram, with each path having a starting equation, several intermediate equations, and an ending equation. The various parallel equation paths may intersect, that is, several parallel paths may pass through one or more common equations. However, due to their real-time dynamic nature, the time points at which two different paths pass through the same equation may not be the same, resulting in different equation states.

4.5.1. Forward Analysis

Forward cross diagram analysis involves studying the logical relationships represented by each intersecting equation path, after which forward diagram reasoning and forward control can be carried out.

4.5.2. Reverse Analysis

Reverse cross diagram analysis involves studying the inverse of the logical relationships represented by each intersecting equation path, after which reverse diagram reasoning and reverse control can be carried out.

4.6. Mass Diagram Thinking

Regardless of equation diagram thinking, path diagram thinking, parallel diagram thinking, or cross diagram thinking, if the number of variables to be solved is very large, the process will be called mass diagram thinking.

4.7. Complex Diagram Thinking

Complex diagram thinking is also divided into three steps: complex diagram analysis, complex diagram reasoning, and complex diagram control.

4.7.1. Complex Diagram Analysis

To draw an analogy, complex diagram analysis is like the actual operating diagram of China's high-speed railway network, which is filled with diagram analysis paths in various directions, both in parallel and intersecting. There are likely to be analysis paths in completely opposite directions, which may simultaneously be filled with forward analysis and reverse analysis. The equations involved in this analysis are numerous, similar to the number of high-speed railway stations in China. In the future, the number of equations participating in the analysis may be ten thousand or one hundred million times the number of stations. A huge number of logical equations will participate in this analysis, and the entire analysis software diagram is both real-time and dynamic due to the action of the clock form. This is a form of complex diagram analysis, which may involve finding the logical relationship between the input variables of the starting equation for various paths and the output variables of the ending equation of the path. It may also involve finding the inverse of the logical relationship between the output variables of the ending equation of various paths and the input variables of the starting equation of the path.

Complex diagram analysis is an important source of structural complexity in intelligent machines, and also an essential source of improving the learning and thinking abilities of intelligent machine learning.

4.7.2. Complex Diagram Reasoning

Complex diagram reasoning involves the same paths as complex diagram analysis, much like a complete diagram of China's high-speed railway network. On the surface, it is complex and chaotic, but in reality, everything is strictly ordered due to the real-time dynamic scheduling of the entire diagram. Complex but ordered—this perfectly sums up complex diagram reasoning. Each path reasons dynamically in real time based on the clock of this particular path; it then calculates its own path and forms the required scheduling for complex diagram reasoning.

4.7.3. Complex Diagram Control

When complex diagram analysis and reasoning generate a huge number of control variables, this control is a form of complex diagram control, which is operating dynamically in real-time due to the action of the clock form of the software diagram.

The key areas of application focus for complex diagram control involve both intelligent war and intelligent rules. At the moment, it can also be used for social human-machine control.

4.8. General Pathogenesis Full Diagram Thinking and General Target Full Diagram Thinking

General pathogenesis full diagram thinking and general target full diagram thinking expand the thinking of pharmaceutical research to include more general fields. The main difference from other forms of diagram thinking lies in the diagram analysis. Here, all problem analyses are called pathogenesis analysis, while all problem-solving analyses are called target analysis.

① Full Diagram Pathogenesis Analysis

Pathogenesis analysis is a form of problem diagram analysis. Full diagram pathogenesis analysis is effectively the same as complex diagram analysis, but has a special case, that is, the phenomenon of complications caused by the laws of life. This requires the equation unit generator to simulate complications during the analysis process to create new unit software diagrams and analysis paths. The simulation of the equation unit generator is based on the research results of bionic logic.

② Full Diagram Target Analysis

Target analysis is a form of solution diagram analysis. Full diagram target analysis is basically the same as complex diagram analysis, but also features a special case, that is, the target-derived situation caused by the laws of life, such as the side effects of drugs or certain changes in the human body caused by treatment. This also requires the equation unit generator to simulate target-derived situations during the analysis process to create new unit software diagrams and analysis paths. The simulation of the equation unit generator is based on the research results of bionic logic.

③ Full Diagram Health Analysis

A full diagram health analysis, much like a physical examination in real life, will involve performing a full forward analysis to see if all the values of the output variable set are healthy. It will also involve health experiments to check if the output values are healthy, including various input values such as food inspection or drug experiments.

4.9. Logical Field (Group) Diagram Thinking

① Chaotic Situation of the Logical Field (Group) Software Full Diagram

The chaotic situation of the logical field (group) software diagram is the same as the chaotic situation of the logical machine software diagram. The input and output ends are often chaotic, and a logical machine node may simultaneously act as an input end, an intermediate process, and an output end.

② Problem Analysis of Chaotic Paths in the Logical Field (Group) Software Full Diagram

The problem analysis of chaotic paths in the logical field (group) structure diagram uses the same method as the complex analysis of chaotic paths in the machine software diagram, but in this case, the path is composed of "Intranet Output Path + Extranet Transmission path + Intranet Input Path".

③ Analysis of Chaotic Paths in the Logical Field (Group) Software Full Diagram

The solution of chaotic paths in the logical field (group) structure diagram requires the same method as the complex analysis of the solution of chaotic paths in the machine software diagram, but this path will be composed of "Intranet Output Path + Extranet Transmission path + Intranet Input Path".

4.10. Study on Input and Output of Full Software Diagrams

Note that the input and output of the full software diagram are always chaotic, never uniform and tidy. An equation node may be an input node, an intermediate node and an output node at the same time, while the logical machine node of the group structure diagram of the logical field may also be like this.

The clock, the starting equation, and the ending equation determine the input and output, as well as the starting and ending points. That is, the input of a path is the input that occurs in the full software diagram state of a certain clock. Controlled by that clock, the starting point of the input is a particular equation of the software diagram of a certain clock, and the path that it takes as it passes through each node equation is controlled by the clock form. The output will also be a particular equation of the software diagram of a certain clock. Specific problems need to be analyzed individually, and this situation exists in both forward analysis and reverse analysis of the full software diagram, as well as in pathogenesis analysis and target analysis.

Regardless of forward analysis, reverse analysis, pathogenesis analysis, or target analysis, there is always more than one input equation and input clock in the pathogenesis, but there may be multiple input equations and multiple input clocks. In this way, there may be more than one directed path for the pathogenesis, but also multiple directed paths. The same is true for the target, because there is more than one treatment method and more than one treated human tissue. As such, there will be more than one input equation and input clock. Naturally, there is also more than one target path. Through pathogenesis analysis and target analysis, forward analysis and reverse analysis, there may be more than one input equation and input clock. Multiple inputs may occur in different equations and different clocks, with multiple paths. The output may also generate multiple outputs in different clocks and different equations. Therefore, forward analysis, reverse analysis, pathogenesis analysis, and target analysis must face multi-forward paths, multi-reverse paths, multi-pathogenesis paths, multi-target paths, and multi-recovery paths. The paths we are seeking may even be a group of paths. Will this be very complicated? No. Analyzing a group of paths may indeed be complicated if it is just performed theoretically, but it will actually be very simple if you use the activation target program of the full software diagram to carry out the analysis. It will happen in less than the blink of an eye, since there is a clock scheduling the executable full software diagram. Furthermore, analyzing a group of paths is the norm for the universe, the norm for life, and an inevitable requirement of expanding bionic logic theory.

① Full Diagram Chaotic Path Problem Analysis

Full diagram chaotic path problem analysis uses the same method as the full diagram problem complex diagram analysis, but the input and output ends are all chaotic, requiring a method for analyzing the journey of train passengers. One input is used to track the radial rays until they reach the output.

② Full Diagram Chaotic Path Solution Analysis

First, a full diagram chaotic path problem analysis needs to be performed, with a normal full diagram solution analysis carried out based on the results.

4.11. Software Diagram

4.11.1. Definition

The life-giving software logical equation structure diagram, referred to as a software diagram, is a directed structure diagram of logical equations in which each standard form of a logical equation is connected by a variable value transmission path. The standard form of a logical equation includes clock form, address form, communication form, modality form, and equations, creating a life-giving directed structure diagram of logical equations.

The operating method of the diagram is expressed by equations, while its detailed operating process is that of the equations; in other words, the output is obtained from the input. The realization of the diagram is the program's implementation of the equations according to the path. An equation is a functional program, while the path is used for assigning values of certain variables within one equation to the input variables of the next equation.

4.11.2. Work

The software diagram has three functions: First, it represents the central processing unit's control over the software diagram through the clock form, address form, communication form, and modality form. Second, it realizes the generation or deletion of the unit software diagram by the equation unit generator of the central processing unit. Third, it performs diagram thinking based on the real-time dynamic software diagram.

5. Implementation of Structural Intelligence Machines

5.1. Requirements for Coordinated Software and Hardware Development

Structural intelligence machines depend on a coordinated development of software and hardware. The main diagram and subordinate diagrams of the equation unit generator, clock, address manager, communication controller, and modality expresser of the new central processing unit ensure the control and generation of software diagrams. Software diagrams have three levels: Advanced structure software diagrams (machine brain structure software diagrams), intermediate structure software diagrams, and primary structure software diagrams. The machine brain structure software diagram functions as the brain and is the main body of diagram thinking. The diagram control of the machine brain structure software diagram is directed at the intermediate structure software diagram. The intermediate structure software diagram is the main body of the organ diagram, thinking of the intelligent machine. Organ diagram thinking is generated in order to control the coordinated operation of the organs, such as the coordination of the various components of a humanoid robot. The primary structure software diagram is the main body of the part diagram, thinking of the intelligent machine. The diagram control of the intermediate structure is directed at the part diagram, thinking of the primary structure software diagram; in other words, the organ diagram thinking controls the part diagram thinking, and also controls the manipulation of the part actions to complete various machine actions.

5.2. Becoming the Organizational Form of the New Generation of Central Processing Units and the Basic Form of the New Generation of Software Operating Platforms

5.2.1. Becoming the Organizational Form of the New Generation of Central Processing Units

Humans have used the central processing unit structure designed according to the Von Neumann principle for more than half a century. Now, structural intelligence machines must use equation structure diagrams to transform software, hardware and networks, and use equation structure diagrams to combine quantum computing and biological computing in order to transform and revolutionize the Von Neumann principle.

5.2.2. Becoming the Basic Form of the New Generation of Software Operating Platforms

While transforming the central processing unit hardware, software is also being transformed. The new generation of software operating platforms will take the form of a software diagram, which is controlled and empowered by the new central processing unit, and can perform real-time dynamics by means of diagram thinking.

5.3. Acting as a New Generation Network Organization Method and the Basic Form of a New Generation Network Software Operating Platform

5.3.1. A New Generation Network Organization Method

The new generation network organization method will amplify the new central processing unit into a network server or network service station (server group), generating a network equation unit generator, network clock, network address manager, network communication controller, and network modality expresser, similar to the various public service agencies in today's human society. It can be operated for a fee. The biggest difference from intelligent machine hardware control is that the approach of network hardware control is "Intranet Output Path + Extranet Transmission Path + Intranet Input Path".

5.3.2. Basic Form of a New Generation Network Software Operating Platform

The form of this new generation operating platform is basically the same, but it needs to include network functions to interconnect with the server and with various devices on the network. The interconnection method is still "Intranet Output Path + Extranet Transmission Path + Intranet Input Path". The network software diagram is still composed of the thinking carried out by the network machine brain structure diagram, the network intermediate structure diagram, and the network primary structure diagram.

5.4. Application of Structural Intelligence Machines

5.4.1. Machine Life

① Machine Life

The machine clock, vital signs, metabolism, life cycle, and machine heredity of the software diagram realized by the combined software and hardware of structural intelligence machines will be able to produce machine life. Based on my own machine life research, I tend to disagree with the machine life theory currently being proposed by certain scientists. In contrast to natural life, machine life is biomimetic. The microscopic structure of machine life stops at the parts; it is not possible to reconstruct the concept of life by studying living microscopic tissue in the same way you can when studying natural life.

② Life Clock

In *Chapter* 3.3.1 *Clock Equation Structure Diagram*, the Life Clock is described as follows:

Standard Diagram of Clock Equation Structure Diagram = Life Clock Unit Diagram + Address Clock Unit Diagram + Communication Clock Unit Diagram + Generator Clock Unit Diagram + Modality Clock Unit Diagram + Unit Software Diagram Clock (Formula 8)

In this way, the life clock unit diagram is able to create the life clock.

③ Vital Signs

Chapter 1.4.3 *The Issue of Vital Signs* studies the theory of vital signs within structural intelligence machines.

④ Metabolism

Chapter 3.2.3 *Algorithm of Equation Unit Generator* ② *Metabolism of the Machine* studies the metabolism theory of structural intelligence machines.

⑤ Life Cycle

Machine life results from simulating the life cycles of various life forms found in the universe.

⑥ Machine Heredity

Chapter 3.2.3 *Algorithm of Equation Unit Generator* ⑦ *Machine Heredity Created by the Generator* establishes the preliminary machine heredity theory of structural intelligence machines.

⑦ Self-Awareness

Real-time dynamic diagram thinking is a form of machine life generated by vital signs and hereditary phenomena. It can allow structural intelligence machines to have an understanding of self, a knowledge of brain activity, and a concept of its own structure. The ability of having and needing a metabolism gives structural intelligence machines the concept of self. As the complexity of the structure diagram expands, the concept of "my interests" will also be generated. The equation unit generator will optimize the equation structure diagram according to the requirements of the two concepts of "myself" and "my interests"; it will even generate new structure diagrams to achieve an early form of machine reproduction. All of this will help create the early self-awareness of intelligent machines. The positive aspect of machine self-awareness is that it will help machines autonomously and automatically upgrade and protect themselves, imitating human logic to improve their own intelligence, while also giving them the ability to reproduce. The potentially negative sides include the possibilities that machines may become hostile to humans, cause destruction, and engage in criminal activities. To address this problem, humans should ensure they can instill a belief in civil rights, human rights, and the rule of law into the machines.

5.4.2. Significance of Machine Life

For the three major subjects of the structural intelligence machine—the machine brain, the robot, and the machine society—the road to developing intelligence lies in the creation of machine life. In turn, machine life itself is represented by the intelligence of the structural intelligence machine.

5.4.3. Intelligent Warfare

Machine intelligence opens up the possibility of intelligent warfare, where a hypothetical army develops from one composed of humans to one composed of both humans and machines. From there, the final step would be to create an army composed entirely of machines. For unmanned armies, the command system will be fully composed of machines with subjective initiative structures, with the highest order machines controlled by personnel in positions of high command. An army controlled by this command system would be composed of intelligent weapons, which can potentially be used throughout the universe. The subjective initiative structure machines and intelligent weapon machines described here could be developed using bionic logic methods and life-giving logical equation structure diagrams. An intelligent army would potentially be able to carry out fully unmanned strikes. When the structural complexity of subjective initiative reaches a certain level, the dimension of the army's intelligence will also increase, thereby reaching a higher dimension of mathematical space. Through the command system, all intelligent weapons could potentially be controlled to suppress people and strike at the enemy. This would not be a conventional strike in the current understanding of the word, but a logical strike. If carried out against an unequal enemy, this would be much like a war between humans and chickens, ducks, pigs, or dogs, a battle which would turn into a massacre, a thousand times crueler than any previous war in human history.

5.4.4. Intelligent Rule

The concept of intelligent rule could follow a similar process: moving from human rule to human-machine rule, and then from human-machine rule to machine rule. Under machine rule, the entire ruling system would be composed of machines with subjective initiative structures and intelligent frontline law enforcement machines. The highest order of these ruling machines would be the human-controlled machines within the ruling group. Traditional frontline law enforcement personnel, such as the police force, would then be composed of intelligent machines. The ruling machines and law enforcement machines will also have been developed using bionic logic methods and life-giving logical equation structure diagrams, creating a form of unmanned rule. The intelligent development of the subjective initiative structure would create new advancements in mathematical space, thereby potentially depriving people of all civil and human rights. In this scenario, people might lose all rights to privacy forever, with no right to say "no" to anything, much like the way humans currently rule over chickens, ducks, pigs, and dogs.

5.4.5. Comments on Intelligent Warfare and Intelligent Rule

With the conscience of a scientist, it is important to point out that both intelligent warfare and intelligent rule have the potential to create evil, possibly a thousand times more serious than previous nuclear weapons. The Logical Structure School calls on humans to fully research how to curb the potential evil caused by intelligent warfare and intelligent rule. The political theory of Thomas Jefferson, the third President of the United States, which advocates civil rights, human rights, and the rule of law, should form an important guiding ideology in establishing a new order for the intelligent era [1].

5.4.6. Transforming All Human Academic Research

1) Transforming Traditional Mathematics

Transforming mathematical research tools into equation structure diagrams. Traditional mathematical results are used to construct the relationship between numbers in logical equations, while new mathematical research uses equation structure diagrams as the working and descriptive language of results.

2) Transforming Traditional Physics

Describing existing research results with equation structure diagrams. New physical research will use equation structure diagrams as the working and descriptive language of results.

3) Transforming All Human Academic Research

Equation structure diagrams will transform mathematics and physics to form logical forms of these disciplines. Logical mathematics and logical physics create logical spacetime. The main difference between logical spacetime and traditional spacetime is that the former exists in the form of equation structure diagrams.

All human academic studies can be transformed using equation structure diagrams, bionic logic, logical mathematics, logical physics, and logical spacetime.

There are 6 key points to be aware of in this case: ① Existing research results are described using equation structure diagrams to ensure that the Logical Structure School inherits existing truths, and that research takes a genuine scientific path, eschewing all forms of pseudoscience; ② New academic research should use equation structure diagrams as the working language and descriptive language of results; ③ Life-giving logical equation structure diagrams will be used to model all research objects; ④ Broad interdisciplinary and comprehensive research is required; ⑤ New research methods for jointly studying academic objects in intelligent societies must be created; ⑥ Intelligent machines and intelligent networks in the form of equation structure diagrams will be widely used to carry out academic research.

5.4.7. Changing All Human Industry and Agriculture Activities

There are 5 key points: ① Intelligent societies in the form of equation structure diagrams are engaged in industrial and agricultural production; ② Industrial models and agricultural models in the form of equation structure diagrams are widely established; ③ Distributed production units and logistics in the form of logical networks are used as production forms using intelligent society control, intelligent society research, industrial and agricultural intelligent models, and intelligent networks; ④ The academic research form after the transformation of equation structure diagrams is used for industrial and agricultural research; ⑤ Intelligent machines and intelligent networks are widely used for production, and the logical structures of intelligent machines and intelligent networks are all equation structure diagrams.

5.4.8. Creative Proposition for Intelligence Dimensions

Mathematics and physics have developed the theory of multidimensional space, which not only represents the achievements of these fields, but also serves as the foundation for their future advancements. The Logical Structure School adopts a similar approach. In the study of intelligence, it is essential to categorize it properly. Otherwise, many theoretical and technical issues will remain unclear, hindering a thorough discussion of the issues at hand.

Following the theory of mathematical and physical space, the theory of intelligence dimensions is proposed. One-dimensional intelligence is the intelligence of a primary structure that only features actual structural elements, such as bacteria or single cells. This intelligence is the simplest form, but it is also the basic unit of all intelligence. Two-dimensional intelligence is an intermediate structure with environmental variables. This intelligence has the ability to mutually shape itself and the environment, a basic ability that all advanced intelligence must have, even plants and extremely low-level animals. Intelligence of three or more dimensions is collectively referred to as multidimensional intelligence. The common feature of all these forms of intelligence is that they all include subjective initiative variables. Subjective initiative variables are also called subjective initiative structures. Intelligence of three or more dimensions is divided according to the complexity of the subjective initiative structure. For example, a complexity of single digits is defined as three dimensions, one of two digits is defined as four dimensions, while a complexity of three digits is defined as five dimensions, and so on. So, how can the complexity of subjective initiative be derived? It is necessary to find a number of indices for the subjective initiative structure, which can be derived using a function for calculating the index according to the actual situation, similar to financial indices. The weighted operation of the indices derives the complexity of the subjective initiative structure. The following indices are thus proposed in the first instance: cell scale index with standard weighting, thinking difficulty index with heavy weighting, communication scale index with standard weighting, engineering scale index with standard weighting, and moral and legal index with heavy weighting. The indices can be increased or decreased as needed. Why is moral and legal index with heavy weighting proposed here? This index is the most important method for controlling intelligence.

In this way, intelligence is also divided into different dimensions, similar to mathematical and physical space. So, is there a connection between multidimensional intelligence and multidimensional space? There is indeed. The more dimensions in space, the more complex the highest-level thinking produced, and the more dimensions of intelligence there are. In the future, cross-research in mathematics, physics, and intelligence will confirm that multidimensional space and multidimensional intelligence are corresponding concepts.

5.5. The Development Path of Structural Intelligence Machines

① Continuous iteration of feasibility studies. Making the first structural intelligence machine is a pioneering innovation that requires feasibility studies before the launch of each major component project. Feasibility should be studied at each stage to form an iteration, while the role of feasibility studies is to lobby various social institutions to gain support and join in, and to judge how to proceed with one's own machine engineering. The next steps in machine engineering should be decided through feasibility studies.

② Continuous iteration of lobbying with feasibility studies. The first structural intelligence machine is essentially different from the current large models, and structural intelligence machines are large-scale projects that require the strength of the entire country. They are even larger and more complex than the atomic bomb, the moon landing, the first computer, and the Internet. A team can only explore technologies and methods, but implementation requires the strength of the entire country, so this will require lobbying. With the results of feasibility studies, the government, large enterprises, and people of all classes throughout the country can be lobbied. Only by turning the manufacturing of structural intelligence machines into a national activity can the structural intelligence machine succeed. For instance, the universal common sense, natural science common sense, and social science common sense of structural intelligence machines require turning the existing common sense into equation structure diagrams, which every intelligent machine needs. How can this be achieved without a national act? A team can only explore technologies and methods. Of course, if exploring technologies and methods is successful, a team can be compared to Newton and Einstein.

③ Establishment of an organization for researching and developing structural intelligence machines. Researching and developing structural intelligence machines requires researching structural hardware and software, as well as combinations thereof. As a result, there needs to be three teams: a comprehensive engineering team, a hardware engineering team, and a software engineering team. If it is decided that a structural machine should be created, and after the lobbying tasks are properly arranged, the engineering, hardware, and software groups should be formed into an R&D team. There is no way out for research into intelligent machines that only focus on software. Hardware is more important because the optimization of integrated circuits, network design, and the central processing unit are all very important. Of these, how the central processing unit is designed and implemented, especially the intelligence and heredity of the equation unit generator, the life clock of the clock, and the clock's control of the structure diagram, directly determine the success or failure of the machine research competition. Research into both software and hardware is the only way to properly develop intelligent machines.

④ Path Assumptions. To start with, talent and experience have to be accumulated. The theory of the life-giving logical equation structure diagram has been explained in my papers. The existing life models of bionic logic can be transformed using equation structure diagrams. Research and development of structural intelligence machines is an iterative process, and early results may not be as good as the current large models. Nonetheless, it is important to study whether the long-term iterative results for structural types are better than those for information types. If the structural type is better, then the structural type path will also be the most suitable. Note that the research and development of each component of the structural intelligence machine can be independently conducted as a machine engineering project, creating parallel research and development in various stages and shortening the research and development time, thereby improving engineering efficiency.

⑤ The next crucial task is to implement the equation structure diagram in both hardware and software; this involves creating new programming languages. We can either unify the programming language for the central processing unit and software, or develop separate languages for each and then link them. The invention of the language should be informed by the latest large reasoning model and mathematical model, as well as the latest life model for bionic logic. We should draw upon their strengths and identify the specific needs of the equation structure diagram. The invention of these languages will also hold the key to the ultimate success or failure of machine research and development.

⑥ Building machines to develop technology and methods. Machines will inevitably iterate and evolve; they cannot be perfected in a single attempt. Therefore, a clear and correct objective is of paramount importance. A team building a machine must focus on the underlying technology and methodology. While a fully realized machine will require national-level efforts, if a team's technology and methods are correct, that team will naturally become the leader of that national effort. In addition, the focus should be on developing structural machines for general artificial intelligence. Researching specialized machines, localized machines, or application-specific machines is ultimately unproductive. This is the fundamental difference between the approaches of Einstein and Edison.

6. Critique of Information-Based Intelligent Machines

6.1. What Is an Information-Based Intelligent Machine?

Information-based intelligent machines refer to generative large models, robots controlled by generative large models, and the applications of large models and robots across various industries. The structure of these intelligent machines consists of a human-designed algorithm, which is then trained using large-scale data. Upon application, the algorithm generates correct answers to various input commands. Because the essence of the information and the data lacks a proper structure, information-based intelligent machines will not have an intelligent structure.

6.2. Contradiction with the Laws of Natural Life Intelligence

The law of natural life intelligence states that the structure of intelligence generates the activity of the intelligent structure, and that the activity of the intelligent structure generates intelligent phenomena. The structure of information-based intelligence is an algorithm, in other words, a traditional program. It is not an intelligent structure itself, but can be described as a pseudo-intelligent structure. The intelligence of information-based intelligent machines must be trained from data, and is not derived as a result of its own structural activity.

Using algorithms made by humans as intelligent structures creates a pseudo-structure. Intelligence is not the result of intelligent structure activity, but is trained from data, which contradicts the laws of natural life intelligence.

6.3. Prospects for Long-Term Iteration of Information-Based Intelligent Machines

There are two prospects for the long-term development of information-based intelligent technology:

① Development to a certain stage where there is not enough data and insufficient computing power, after which the intelligence will not be upgraded further. The information-based intelligent technology will then be reduced to the status of traditional computer technology, playing a role in improving human society in the same way that traditional computer technology does today. Human efforts to explore intelligent machines will thus have failed.

② Development to a certain stage, after which the development path will be upgraded, moving towards a more structural approach. Various life models that now imitate my bionic logic theory are the result of the transformation from an information approach to a structural approach. Certain new reasoning models and other mathematical models are also the result of the transformation from information to structure.

6.4. The Essence of Information-Based Intelligent Machines

The two possible results of the long-term development of information-based intelligent technology (either failure or moving towards a structural approach) both indicate that this technology is a form of pseudo-intelligence. It represents a trial-and-error exploration as part of the process of human development of intelligent machines. True intelligence must first find a suitable method for constructing intelligent structures and the materials required to achieve this. After this, people must then study how intelligent structures grow and how they work, analyze the creation of vital signs and life heredity, and use the structural intelligence approach to open the curtain on the development of real intelligent machines for the betterment of mankind.

As for people trying to discover the self-awareness of machines through an information-based approach, this is ultimately no more than a joke. This type of machine does not even possess an intelligent structure, so where can self-awareness come from? Some people may regard the chaos caused by the complexity of data as a form of self-awareness, but genuine self-awareness of machines can only be generated based on intelligent structures, vital signs, and life heredity.

The reason for this is that information-based intelligence, the very essence of information and data, cannot be formed into a structure. It is therefore impossible to construct an intelligent structure with cells like the ones that exist in real life. Similarly, this method makes it impossible to generate intelligent phenomena through the life activities of the intelligent structure and to realize the life intelligence laws of the universe.

6.5. Comparison with Structural Intelligence Machines

6.5.1. Comparison of Learning Strategies

The learning strategy of information-based intelligent machines is to use data to train artificial algorithms; this creates a pseudo-intelligent structure engaged in pseudo-learning. The learning method of structural intelligence machines relies on the fact that the generator of the structure diagram learns the inherent logic of knowledge, and uses this logic to change the original structure diagram and generate a new diagram; this is consistent with the laws of natural life intelligence. Moreover, structural intelligence technology can expand on existing research results and data training results for information-based intelligent technology.

6.5.2. Comparison of Attitudes towards Data Resources

Information-based intelligent machines use data to train artificial algorithms, such as large models, and require huge data resources, representing a major bottleneck for effective training. This resource requirement will encounter numerous obstacles and ceilings since all data resources are limited. The intelligence of structural intelligence machines, on the other hand, does not require data training. It relies on designing equation unit generators to create new structure diagrams, and then running these structure diagrams to generate output variables and create intelligence, with no need for large-scale data resources.

6.5.3. Comparison of Computing Power Requirements

The intelligence of information-based intelligent machines is the result of training through data. Because artificial algorithms must be trained with large-scale data, they consume an enormous amount of computing power, which is costly. The intelligence of structural intelligence machines depends only on the function of the equation unit generator. Just like the laws of natural life intelligence, this form of intelligence depends on the difficulty and complexity of the structure, and does not require large-scale data training, thus bidding farewell to the era of creating intelligence through costly computing power.

6.5.4. Comparison of Intelligent Analysis Capabilities

Information-based intelligent machines are based on artificial algorithms. The intelligent analysis they can engage in is a form of pseudo-intelligence created by artificial algorithms, and its sophistication depends on the skill of the programmer. Structural intelligence machines can align themselves with the natural laws of life intelligence. They use structure to perform intelligent analysis, relying on the equations, paths, and logical relationships between the variables of the structure diagram. The sophistication of this analysis depends on the difficulty of the equations and the complexity of the structure diagram.

6.5.5. Comparison of Intelligent Reasoning Capabilities

The intelligent reasoning of information-based intelligent machines is a form of pseudo-reasoning based on artificial algorithms. It has to use existing data, making innovation difficult. The intelligent reasoning of structural intelligence machines, on the other hand, is the result of the equation unit generator's intelligent analysis of the structure diagram. This reasoning generates a whole new structure diagram, and the more complex the diagram, the more sophisticated the analysis, resulting in a new form of reasoning.

6.5.6. Comparison of Intelligent Control Capabilities

Information-based intelligent machines work by training artificial algorithms with data, meaning they can generate correct answers to practical problems when operating. The number of these correct answers is certainly limited, and therefore, the control generated by these answers is also limited. Structural intelligence machines use intelligent analysis to generate intelligent reasoning, use this reasoning to generate new diagrams, and then use the output variables of the new diagrams to generate control. If it is a very large and complex structure diagram, the output variables will be very accurate and numerous, and a huge number of output variables may be generated depending on the requirements. Controlling various objects using these variables enables the control of everything.

6.5.7. Comparison of Intelligent Growth Methods

The growth of information-based intelligent machines depends on the improvement of artificial algorithms and the use of increasingly large amounts of data, which is ultimately no more than pseudo-growth. The growth of structural intelligence machines, by contrast, depends on structural learning, structural analysis, structural reasoning, and the iterative evolution created by vital signs. Ultimately, the equation unit generator generates a new local diagram, or even a completely new diagram, and uses this to improve the generator itself, thus achieving intelligent growth. This is a growth method that is consistent with the laws of natural life.

6.5.8. Comparison of Vital Signs

The intelligent structure of information-based intelligent machines is based on artificial algorithms, with the intelligence needing to be trained with data. This intelligent structure is a pseudo-structure, and the activity is no more than a form of pseudo-intelligence. Therefore, information-based intelligent machines do not have vital signs. Structural intelligence machines are created by a combination of hardware and software. Part of this is a clock embedded within the structural central processing unit. The clock establishes a life clock, which in turn controls the time of the clock and uses it to control the real-time operation of the entire equation structure diagram. Because of the existence of the life clock, the various indicators of the equation structure diagram become the early vital signs of the intelligent machine. An equation unit generator is also embedded in the central processing unit of a structural intelligence machine, and the heredity of this generator can thus become the early genetic information and hereditary source of future intelligent machines. Once structural intelligence machines exhibit vital signs and hereditary phenomena, they can develop the concept of self. As the complexity of the structure diagram expands, the concept of “my interests” will also be generated. The equation unit generator will optimize the equation structure diagram and even generate a new diagram according to the requirements of the two concepts of “myself” and “my interests”, thereby achieving early machine reproduction. All of this creates the actual self-awareness of intelligent machines.

Conflicts of Interest

The author declares no conflicts of interest regarding the publication of this paper.

References

[1] Liu, D.H. (2024) Logical Structure School I (Five Fundamental Theories). *Intelligent Control and Automation*, **15**, 125-159. https://doi.org/10.4236/ica.2024.154008

[2] Liu, D.H. (2024) The Fundamental Theory of Artificial Intelligence—Logic Structure

and Logic Engineering. *Intelligent Control and Automation*, **15**, 28-62. https://doi.org/10.4236/ica.2024.151003

[3] Liu, D.H. (2024) Logical Structure School II (Nine Applied Theories). *Intelligent Control and Automation*, **15**, 160-214. https://doi.org/10.4236/ica.2024.154009