Artificial Intelligence-Driven Construction of Virtual Experimental Environments for Information Technology Teaching

Zhiqiang Wang*

Jiyuan Vocational and Technical College, Jiyuan, 459000, China
**Corresponding author: 0000217@jyvtc.edu.cn*

Abstract: *The problems of the traditional virtual simulation environment for teaching experimental science with information technology are context rigidity, lack of interaction and high cognitive load. To solve the problems above, we have put forward a framework for building an intelligent educational form-driven virtual experimental environment, and the three parts of this framework are intelligent educational forms, multi-agent computational models and self-evolving semantic associations. First, we will construct virtualized representations and multimodal interaction modes to clarify the generation mechanism of dynamic experimental scenarios and the spatial logic for adaptive regulation of cognitive load; second, based on the concept of multi-agent cooperation, we will build a computational model to achieve behavioural modelling of experimental objects, evolve state graphs driven by heterogeneous data, and establish a reinforcement learning framework for intention recognition and feedback generation; finally, we will design a self-evolution mechanism for the virtual experimental environment that integrates dynamic resource organisation according to domain knowledge graphs, adaptation of virtual instrument operation logic via generative adversarial networks, and semantic annotation and path optimisation for multimodal interaction trajectories. The framework above is now a complete system of theory and technology for virtual laboratory environments in information-technology education.*

Keywords: *Artificial Intelligence, Virtual Experimental Environment, Information Technology Teaching, Multi-agent Collaboration, Knowledge Graph, Generative Adversarial Network, Cognitive Load Regulation*

Introduction

Information-technology experimental teaching can help students develop computational thinking and engineering literacy in the process of learning. Most of the current virtual experiment environments have fixed scripts and interaction interfaces, so it is not very simple to customise them according to different cognitive levels or changes in the demand for inquiry. With the development of artificial intelligence technology, a new type of virtual laboratory environment that is context-aware, adaptive and self-optimising has begun to attract the attention of scholars. The reasons for this study are as follows: First of all, intelligent virtual representations and interaction forms can overcome the spatial and temporal constraints of physical experiments, and we can extend the scale and depth of experimental teaching. Second, in order to improve the efficiency of teaching and learning in the experiment, multi-agent cooperation and reinforcement learning can be employed to study students' motivation and provide early feedback in a virtual environment. Third, to construct an autonomous evolution framework based on knowledge graphs and generative adversarial networks, it will be possible to dynamically allocate experimental resources and automatically adjust the operating logic to reduce the maintenance cost of the environment. The above studies have shown that an intelligent transformation of the experimental teaching environment in information technology is necessary, and they have also provided some theoretical support for this.

1. Intelligent Educational Forms and Cognitive Architecture of Virtual Experimental Environments

1.1 Virtualised Representation and Interaction Modalities in Information Technology Experimental Teaching

Virtualisation of information-technology experimental teaching changes the experimental objects, processes and results that can only be obtained by means of physical devices into computable and reconfigurable digital entities through symbolic abstraction, geometric mapping and dynamic data visualisation. The System of Representation consists of topological descriptions of the experimental components, sequential logical expressions for signal flow, and parametric models of state variables; thus, the experimental elements can be developed independently in virtual space without physical constraints. The Size and Range of the Virtual-Reality Display will affect the depth of students' learning about the reasons for the experiment; it should be neither too general nor too specific[1].

The Construction of the Interaction Mode is based on multimodal perception fusion, has a vision-dominant operation window, auditory event alerts and haptic force feedback. The interaction mode of the virtual experimental environment should be that of command input and result output; to enhance the sense of presence and operational accuracy for learners, redundant coding and complementary mapping should be added in all modes. For example, we can see that the light effect produced by switching a circuit from off to on has changed; at the same time, we will also hear a short click sound of a relay closing, and both occur at the same time to form a joint causal experience. The Design of Interaction Modes should be based on the principle of dual coding; that is, the same experimental operation should produce consistent but not identical information in all sensory channels to reduce ambiguity and improve the identifiability of representations. The size of the time-window tolerance for multimodal alignment affects perceptual consistency; it should not be too small, or the modalities will be out of sync, and it should not be too large, or causality will be lost. In desktop virtual environments that do not support physical force feedback, vibration coding is employed to generate haptic feedback; that is to say, all kinds of vibration patterns are mapped to different experimental situations to create a weak but effective sense via the haptic channel.

1.2 Artificial Intelligence-driven Generation Mechanism for Dynamic Experimental Scenarios

Generation of dynamic experimental scenarios is based on the model of a triad consisting of learner states, experimental goals and domain knowledge. According to the sequence of operations and the results of the previous learning, the artificial intelligence system will construct an implicit cognitive trajectory vector and use this vector as the start point for scenario generation. Based on the current learning needs and within the limits of known knowledge points and difficulty levels, a conditional variational autoencoder or a diffusion model can be employed to generate experimental initial conditions, fault injection locations and ranges for parameters, etc. This generation is not random; it should meet the causality consistency requirement of experimental logic, and the generated scenarios must be solvable and pedagogically sound in terms of physics or algorithms.

Dynamism in Scenario Generation is a kind of closed-loop control. After the learner finishes a set of operations in the current experimental scenario, the system will collect the corresponding behaviour data (such as operation duration, types of errors, steps for parameter adjustment), feed it back into the conditional space of the generation model, and so on. Then, according to the teaching strategy network, the model will change the difficulty and direction of the next scenario. Therefore, we can avoid the fixed defects of fixed experimental scripts and bring the experimental situation closer to the learner's zone of proximal development. At the same time, there is an internal submodule in the generation module for scenario validity checking that filters out candidate scenarios that violate the fundamental laws of experimentation or contain logical contradictions; thus, the dynamic generation process will always be within the teachable range[2].

1.3 Spatial Logic for Adaptive Regulation of Learners' Cognitive Load

To reduce the cognitive load of learners in the spatial logic of the virtual experiment, the following three modifications are made: Scene Organisation, View Position and the Quantity of Information. The Layout of the Space will be based on how often it is used and what functional requirements the group experimental instruments, control panels and data display areas have; frequently used interactive elements will be in the main window, and auxiliary reference data can be folded away in a sidebar. If the system finds that a learner's eyes are moving frequently among many places, it can organise the corresponding areas in two-dimensional or three-dimensional space through spatial remapping to reduce the extra cognitive load of visual search and mental rotation.

Adjustment of information density is based on how much working memory is available now. Based on changes in the eyes or changes in entropy during operation, determine how much new information the learner can currently process and adjust the number of parameter-adjustment sliders, waveform channels or the level of detail in the status prompt text shown in the virtual environment accordingly. When the occupancy is low, all the experimental elements will be shown in the spatial logic to motivate inquiry-based discovery; when the occupancy is high, only the interaction entry points of non-core components that are needed for the current task will be displayed. According to the principle of progressive disclosure, not all the learning materials need to be presented at the beginning of the regulation process; rather, in order to reduce cognitive load, one can dynamically change the spatial visibility, move irrelevant information to the periphery of vision, or delay the loading of such information.

2. Computational Model of Virtual Experimental Environment Based on Multi-Agent Collaboration

2.1 Behavioral Modeling of Experimental Objects and Design of Agent Rule Base

Behavioral modelling of experimental objects abstracts the physical entities in the virtual environment (such as circuit components, optical lenses and mechanical linkages) into agent units that have state variables, input-output interfaces and behavior functions. All agents have their own physical laws or logical constraints, such as the voltage-current characteristic of a resistor element and the truth table mapping of a logic gate. Behavioural Modelling is the combination of a hierarchical finite-state machine and continuous differential equations; the discrete part is used to handle mode switching in the experiment (such as switch closure and gear adjustment), and the continuous part is employed to describe the analogue evolution of signals. A Port-Connector Model is employed to connect the agents, so the transmission of energy or information will be conservative, and there will be no non-physical causal anomalies[3].

The Design of the agent rule base is organised according to the granularity of knowledge points in experimental teaching, and it consists of production rules and constraint conditions. The three kinds of rules in the rule base are: basic physical property rules (such as the encoded form of Ohm's Law and Kirchhoff's Laws), operation response rules (such as the continuous change in resistance due to the rotation of the slider of a rheostat), and fault simulation rules (such as injection methods for open circuits, short circuits or parameter drift). The construction of the rule base is done in an ontological manner; the basic concepts of information technology experiments (voltage, current, frequency, duty cycle, etc.) and their relationships are defined as a directed graph, and rules serve as edge functions connecting concept nodes. There is also an extension interface for the rule base, and given the changes in the organisation of the experimental content at all levels, new component definition files or custom rule scripts can be added.

2.2 Heterogeneous Data-driven Evolution of Experimental Process State Graphs

The Development process state graph of the experiment is a directed graph that shows the path of system states in the virtual experiment. Nodes in the state graph are the global state snapshots of the experimental environment, such as the current values of parameters, connection topologies and operating modes of all agents, and directed edges are labelled with operation events or time advancement steps that cause a change in state. Heterogeneous Data Sources refer to learner operation logs (discrete event sequences), sensor simulation data (continuous signal waveforms) and intermediate variables obtained from rule-based reasoning. The Data Fusion Layer synchronises the time and semantics of all signals to create a single stream of state-transition events, and then these events are used to expand the state graph hop by hop from the start node.

Dynamic Management of the Evolution Process is to construct and prune an incremental graph. The system will construct the state graph at the time of experiment and not list all possible paths beforehand. When a learner performs an operation, the system will check whether the outgoing edges of the current state node already contain a transition target with the same operation label; if such a target exists, it will directly jump to it; otherwise, it will call the simulation engine to calculate the new state, add a new node and edge. To prevent the infinite expansion of the state graph, determine how close the new state is to the existing nodes in the history (e.g., Euclidean distance or structural hashing), and if this distance is smaller than a certain threshold, merge it with the same node based on equivalence class merging. The structure of the built-state graph can be used to log the experiment, and at a later time, one can backtrack or display the path of operation.

2.3 Reinforcement Learning Framework for Intention Recognition and Feedback Generation

The intention-recognition module will map the sequence of a learner's actions to the aim and way of conducting an experiment. This module is a partially observable Markov Decision Process, and the operation sequences are the observation sequences and the intention categories are the hidden states. A Long Short-Term Memory network or a self-attention encoder is employed in this module to obtain temporal features from the history of operations, and then a probability distribution over intentions is produced, such as "tune to the resonant frequency", "verify the parallel current division rule", or "locate a short-circuit fault". Later, according to whether the recognition results are consistent with the feedback from the next operation, the accuracy of intention recognition will be adjusted; thus, one incorrect judgment will not significantly affect the system[4].

The two agents of the feedback generation module are deep reinforcement learning; one is responsible for generating the content, and the other selects the time and mode. The State Space consists of the current category of the recognised intention, the evolution position in the state graph, and historical behaviour statistics of the learner; the Action Space is to generate prompt texts, highlight key components, play demonstration animations, or remain silent. The reward function is based on the operation efficiency index and the degree of deviation from the state graph path; when the learner's next operation is close to the standard path of the recognized intention, a positive reward is given, and a negative reward is given for repeated ineffective operations. Offline training of the reinforcement learning framework is carried out in a pre-built standard experimental path library, and a lightweight policy network is employed for online inference to promptly respond to feedback.

3. Self-Evolving Construction and Semantic Association Mechanism of a Virtual Experimental Environment

3.1 Dynamic Organisation of Experimental Resources Based on Domain Knowledge Graph

The foundation of the knowledge graph in this paper is the knowledge system of information-technology experimental teaching, and concept nodes (such as "analog-to-digital conversion", "sampling theorem" and "clock synchronization") and relationship edges (prerequisite dependencies, analogical mappings and typical applications) have been organised into a directed heterogeneous graph. Based on the concept hierarchy in the course syllabus and the experimental lab manual, a graph is constructed, and then sequential constraints are added to the typical experimental steps to form knowledge paths. All nodes have computable attributes, such as difficulty coefficients, indices of related experimental cases and sets of replaceable components. Graph embedding methods are used to place the nodes in a low-dimensional vector space, and thus, the semantic similarity among them can be expressed as the distance of vectors; this provides a reference for changes in resource organisation[5].

The dynamic organisation mechanism of experimental resources calculates the reachable subgraph and resource invocation weights according to the position of the learner's current operation node in the knowledge graph. A pool of virtual resources will be provided by the system, and among them are an experimental component library, an oscilloscope panel and a signal generator. When the learner has an operational intention, find the corresponding concept node in the graph and expand one or two layers of neighbourhood along the relationship edges to obtain the resource types and invocation frequency statistics of the nodes in this neighbourhood. Based on the above, the resource organisation module puts frequently used resources in working memory and loads infrequently used resources lazily or retrieves them on demand. A mapping relationship between resources and a graph can be used to achieve incremental updates; when a new experimental resource is added, it can be connected to the corresponding concept node in the graph via semantic annotation, and retraining of the entire model does not need to be carried out, so both scalability and simple maintenance of the resource organisation can be achieved.

3.2 Adaptation of Generative Adversarial Networks (GANs) for Virtual Instruments and Operational Logic

Generative Adversarial Networks are used in virtual instruments to solve the problem of operating logic mismatch between fixed instrument panels and different experimental environments. The Generator takes the description vector of the experimental goal (including the required type of measurement, precision range, interaction mode, etc.) as input and outputs the panel layout parameters, types of control components and their response functions for the virtual instrument. According to the operation efficiency data obtained from the actual experiment, determine whether the generated panel meets the conditions for good operating efficiency (e.g., how many clicks are needed to complete a measurement, what is the step error in parameter adjustment by the discriminator). The two are in an adversarial training mode; that is to say, at each time step, the generator is trained to output instrument interfaces according to the scenario, and the discriminator will push the generated results to meet the hidden ergonomic requirements.

Corrections will be made to the causal consistency of the instrument operation logic and the experiment as well. The generator has a differentiable virtual instrument simulation module in its generator, and during the simulation of the output panel in the generation stage, a set of typical operation sequences are injected, and it is verified whether the output readings conform to the corresponding physical laws (e.g., when changing the voltage, it should be observed that the current changes according to Ohm's Law). If the simulation results do not meet the basic physical requirements, a penalty term is added to the loss function, and then the generator corrects the panel parameters or the component wiring logic. After the convergence of adversarial training, the interface of the virtual instrument will be automatically changed according to the different stages of the experiment; for example, it can show the display window of the Bode plot and hide the unnecessary DC range in frequency response tests to reduce operation redundancy[6].

3.3 Semantic Annotation and Path Optimisation of Multimodal Interaction Trajectories

Uniformly represent the operation logs, eye-tracking gaze points, voice commands and parameter adjustment curves generated by learners in the virtual experimental environment as time-stamped heterogeneous event streams in the semantic annotation of multimodal interaction trajectories. The Organisation of Annotations is hierarchical and semantic; at the bottom is the most basic kind of action (selection, rotation, connection and reading), at the intermediate level a sequence of these events is organised into functional modules (building a bridge circuit, measuring gain, etc.), and finally, at the top level, experimental strategies are derived from the sequence of these modules (verify-revise-verify mode or gradual approximation mode). Annotation algorithms are Conditional Random Fields or Temporal Convolutional Networks, and given a small number of pre-annotated trajectories, supervised learning can be employed to automatically perform semantic decomposition of unannotated trajectories.

The Path Optimisation Module takes the annotated semantic trajectory as input, searches for a better operation sequence in the state graph space, and then this will be used by the following Learners. All of the above factors are included in the objective function of optimisation, such as the number of operation steps, time cost and semantic correctness constraints; among them, the correctness constraints are given by the domain knowledge graph (e.g., "the circuit must be closed before measuring voltage" is a hard constraint). PathSearch is a heuristic that estimates the cost from the current node to the end node based on the past statistical data of how long it has taken to reach various places. The best answer is not a single standard answer; rather, clustering can be used to find many equivalent paths under different circumstances (e.g., many routes suitable for various types of cognition). The extended Paths are added back to the knowledge graph as new prior knowledge for the organisation of experimental resources, and thus a closed-loop system of self-improvement has been established.

Conclusion

According to the construction of the artificial intelligence-driven virtual experimental environment for information technology teaching, the whole theoretical system of the three areas mentioned below has been introduced: intelligent educational forms, multi-agent collaborative computational models and self-evolving semantic association mechanisms. For the intelligent educational forms of the virtual experimental environment, we have set out the Design Principles of virtualised representation and Interaction Modalities, revealed the generation mechanism of dynamic experimental scenarios, and built an adaptive regulation logic for cognitive load; for the multi-agent collaborative computational model, we have established behavioral modeling and rule bases for experimental objects, developed a heterogeneous data-driven state graph evolution method, and built a reinforcement learning-based architecture for intention recognition and feedback generation; for the self-evolving construction and semantic association mechanism, we have put forward a dynamic resource organisation strategy guided by domain knowledge graphs, introduced a generative adversarial network-driven adaptation method for virtual instruments, and developed semantic annotation and path optimisation techniques for multimodal interaction trajectories. In the future, we will learn to use multimodal large models to enhance comprehension and generation in experimental situations, improve the environment's ability to understand natural language instructions and complex operational intentions, study cross-scenario knowledge transfer mechanisms to reuse learned strategies and resources from different courses of information technology experiments in a virtual environment, and further optimise the real-time constraints of online inference to reduce computational resource overhead in large-scale deployment.

References

[1] Tang, Y. X. (2026). A strategic study on artificial intelligence technology empowering junior high school information technology teaching. Primary and Secondary School Electrification Education, (7), 46-48.

[2] Yue, X. F. (2026). Analysis of personalized teaching models in university information technology driven by artificial intelligence. Information Systems Engineering, (2), 157-160.

[3] Du, F. Q. (2026). Practical research on artificial intelligence technology empowering information technology course teaching in higher vocational colleges. Computer Knowledge and Technology, 22(4), 138-140.

[4] Xin, P. C. (2026). Application and prospect of information technology in educational reform. In Proceedings of the 2026 Higher Education Development Forum (Volume II) (pp. 241-243). Department of Economics and Management, Inner Mongolia Hongde College of Arts and Sciences.

[5] Yue, X. F. (2026). Analysis of personalized teaching models in university information technology driven by artificial intelligence. Information Systems Engineering, (2), 157-160.

[6] Xu, L. N. (2025). Research on interdisciplinary teaching innovation in information technology driven by generative artificial intelligence. China Information Technology Education, (17), 55-57.