



Article

# Educational Practices and Algorithmic Framework for Promoting Sustainable Development in Education by Identifying Real-World Learning Paths

Tian-Yi Liu <sup>1,†</sup>, Yuan-Hao Jiang <sup>2,3,4,†</sup>, Yuang Wei <sup>2,3,4,†</sup>, Xun Wang <sup>1,\*</sup>, Shucheng Huang <sup>1</sup> and Ling Dai <sup>5,\*</sup>

<sup>1</sup> School of Computer, Jiangsu University of Science and Technology, Zhenjiang 212100, China; 231210704104@stu.just.edu.cn (T.-Y.L.); schuang@just.edu.cn (S.H.)

<sup>2</sup> Lab of Artificial Intelligence for Education, East China Normal University, Shanghai 200062, China; yuanhao.cs.edu@gmail.com (Y.-H.J.); philrain@foxmail.com (Y.W.)

<sup>3</sup> Shanghai Institute of Artificial Intelligence for Education, East China Normal University, Shanghai 200062, China

<sup>4</sup> School of Computer Science and Technology, East China Normal University, Shanghai 200062, China

<sup>5</sup> Department of Education, East China Normal University, Shanghai 200062, China

\* Correspondence: wx\_just@126.com (X.W.); perfectdaring@outlook.com (L.D.)

† These authors contributed equally to this work.

**Abstract:** Utilizing big data and artificial intelligence technologies, we developed the Collaborative Structure Search Framework (CSSF) algorithm to analyze students' learning paths from real-world data to determine the optimal sequence of learning knowledge components. This study enhances sustainability and balance in education by identifying students' learning paths. This allows teachers and intelligent systems to understand students' strengths and weaknesses, thereby providing personalized teaching plans and improving educational outcomes. Identifying causal relationships within knowledge structures helps teachers pinpoint and address learning issues, forming the basis for adaptive learning systems. Using real educational datasets, the research introduces a multi-sub-population collaborative search mechanism to enhance search efficiency by maintaining individual-level superiority, population-level diversity, and solution-set simplicity across sub-populations. A bidirectional feedback mechanism is implemented to discern high-quality and low-quality edges within the knowledge graph. Oversampling high-quality edges and undersampling low-quality edges address optimization challenges in Learning Path Recognition (LPR) due to edge sparsity. The proposed Collaborative Structural Search Framework (CSSF) effectively uncovers relationships within knowledge structures. Experimental validations on real-world datasets show CSSF's effectiveness, with a 14.41% improvement in F1-score over benchmark algorithms on a dataset of 116 knowledge structures. The algorithm helps teachers identify the root causes of students' errors, enabling more effective educational strategies, thus enhancing educational quality and learning outcomes. Intelligent education systems can better adapt to individual student needs, providing personalized learning resources, facilitating a positive learning cycle, and promoting sustainable education development.



**Citation:** Liu, T.-Y.; Jiang, Y.-H.; Wei, Y.; Wang, X.; Huang, S.; Dai, L. Educational Practices and Algorithmic Framework for Promoting Sustainable Development in Education by Identifying Real-World Learning Paths. *Sustainability* **2024**, *16*, 6871. <https://doi.org/10.3390/su16166871>

Academic Editors: Hongjun Wang, Peng Jin, Yanru Chen and Grigoris L. Kyriakopoulos

Received: 1 July 2024

Revised: 31 July 2024

Accepted: 8 August 2024

Published: 10 August 2024

**Keywords:** sustainable development in education; AI for education; educational application; learning path recognition; evolutionary computation



**Copyright:** © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

In recent years, there has been increasing attention on the sustainable development of the contemporary world, with the sustainable development of education being a significant issue. In 2021, UNESCO called for a re-examination of the role of education at a critical time when humanity faces numerous global challenges and societal needs for transformation [1]. The United Nations' Sustainable Development Goals include promoting sustainable and inclusive development of education. Through technological practices in education, sustainable development can be promoted. On one hand, educational technology can facilitate

the global dissemination of knowledge, enhancing educational levels in impoverished areas, such as the ITS intelligent tutoring system [2] and e-Learning technologies [3]. On the other hand, by continuously learning from newly obtained educational data through methods such as data mining [4] and reinforcement learning [5], educational technology is constantly evolving, thereby promoting the sustainable development of the technology itself. Through these two efforts, the sustainable development of education can be ensured.

The field of education is undergoing unprecedented changes and opportunities due to the rapid development of information technology and continuous progress in the social economy. Traditional education models face several challenges, including inadequate levels of personalized learning [6], inefficiencies in knowledge acquisition [7], and unequal distribution of educational resources [8]. Meanwhile, the increasing maturity and widespread adoption of artificial intelligence (AI) technology have introduced new possibilities and prospects to the field of education. Consequently, the pursuit of optimal real learning paths has emerged as a critical research topic in intelligent education. However, achieving the goals of intelligent education necessitates the establishment of a comprehensive learning path system. Learning paths are structured plans that delineate the sequence and steps through which students acquire knowledge and skills. Traditional learning paths often exhibit rigidity and uniformity, failing to meet the personalized learning needs of diverse students. Causal relationships among different knowledge components can generally be represented by causal graphs, with Directed Acyclic Graphs (DAGs) being the commonly used type. DAGs consist of nodes and directed relationships that visually depict cause-and-effect links between knowledge components. Judea Pearl's Structural Causal Model (SCM) serves as a fundamental framework in causal inference research [9]. Graph structure search methods play a crucial role in identifying optimal learning paths for students. Graphs, owing to their ability to effectively represent causal relationships, find applications not only in education [10] but also in healthcare [11], engineering [12], and other domains. Due to the widespread existence of graph structures, they become the backbone of many systems. Graphs store information about entities with interactive relationships as well as specific interaction details. By transforming student learning paths into graph structures and constructing data-driven multidimensional evaluation models that comprehensively consider various indicators such as academic performance, learning behavior, and skill development, it is possible to thoroughly assess students' learning conditions. This approach enables the stable and sustainable development of students' academic careers.

The prevalence of graph structures supports numerous systems by storing information about entities and their interactions. Accurately deriving causal structures from data remains an NP-hard problem, making efficient learning of causal relationships a research focus for decades. Although methods based on deep neural networks, such as Multilayer Perceptrons (MLP), have successfully addressed many real-world problems [13], they suffer from challenges such as poor interpretability [14], slow training speed [10], and inefficiency in adapting to continuously generated new data, necessitating model retraining and resulting in slow adaptation to new information [15]. Recently, KAN [16], with its simpler network structure and stronger mathematical support, has shown promise in replacing MLPs and enhancing deep neural network capabilities further. However, it requires further refinement and optimization for practical applications, along with increased computational resources compared to MLPs. Given the computational demands and interpretability challenges, neural networks were not within the scope of this study. Moreover, score-based graph structure search algorithms effectively identify causal relationships between nodes by employing scoring methods that simplify calculations and can adapt to different scenarios, enhancing the flexibility of graph structure search. However, as the scale of the graph increases, score-based methods become computationally intensive, transforming the graph structure search into a costly optimization problem. For example, the Greedy Equivalence Search (GES) algorithm [17] utilizes the BDeu scoring function for evaluation and employs a greedy search to achieve locally optimal solutions. Similarly, algorithms like the Hill Climbing Algorithm [18] and its improved version, the Max-Min Hill-Climbing

Algorithm [19], involve traversing all adjacent states of the current graph structure, thereby compromising search efficiency.

In contrast, metaheuristic algorithms solve graph structure search problems effectively by avoiding the pitfalls of local optima encountered in greedy searches. For instance, Y. Tian et al. [20] proposed the Multi-Stage Evolutionary Algorithm (MSEA), an innovative approach that divides the optimization process into multiple stages, leveraging the unique characteristics of each stage for targeted enhancement of population diversity and solution quality. This phased strategy effectively mitigates premature convergence of the population, thereby enhancing the algorithm's global search capability. The Golden Eagle Optimizer (GEO) [21], inspired by the multi-stage hunting behavior of golden eagles, demonstrates high flexibility and efficiency during the hunting process. By simulating the eagles' adaptive hunting strategies in phases, GEO optimizes the fitness values of individuals in the population more effectively during the global optimization process. Despite their advantages, general metaheuristic algorithms encounter challenges in sparse graph searches within the Learning Path Recognition (LPR) problem, potentially discarding valuable causal relationships due to inadequate evaluation of adjacent states.

Based on the aforementioned challenges, determining causal relationships among numerous knowledge components and the sparse nature of causal connections between nodes significantly reduce search efficiency for learning paths in the LPR problem. To effectively address this issue, this paper proposes a Collaborative Structural Search Framework (CSSF). Through collaboration among multiple subgroups, individuals in the population are grouped to work towards different objectives, ultimately enhancing CSSF's search efficiency. During the search process, a positive and negative feedback mechanism facilitates the effective identification of high-quality and low-quality edges. The effectiveness of CSSF has been fully validated through experiments on real-world datasets.

## 2. Related Work

### 2.1. Literature Review of Graph Structure Search Based on Causal Discovery

The sustainability of education is a crucial element in fostering students' ability to adapt to future societal changes and competition, promoting technological innovation, and driving social development and technological advancement. Enhancing the sustainability of education lies in cultivating students' learning abilities. By combining students' regular learning data and analyzing their actual learning paths, personalized and tailored education can be achieved, thereby enhancing students' learning abilities and improving the sustainability of education. Causal discovery is a pivotal technique used to infer causal relationships from data, crucial for comprehending the intricate interplay between variables within complex systems. In data-driven research, identifying causal relationships not only elucidates variable interactions but also enables the prediction of future behaviors and forms a foundation for informed decision-making. Graph structure search plays a fundamental role in causal discovery by constructing causal graphs that visually represent the causal relationships among variables. Philosophically influenced by Hume and Kant regarding counterfactual causality, Lewis [22] formalized the counterfactual framework, establishing a comprehensive logical chain for counterfactual causation. His work advanced theoretical studies in causal reasoning and delineated directions for further research. Building on the theory of counterfactuals, Rubin [23] advocated for potential outcomes as potent tools for causal inference, successfully applying them in non-experimental observational studies. This pioneering effort led to the widespread acceptance of potential outcomes [24] as a foundational framework in causal inference. Subsequently, Pearl integrated techniques such as graphical models, structural equations, and counterfactual analysis to propose a novel formal theory of causation. This culminated in the development of Structural Causal Models [25], a widely recognized framework that significantly advanced the integration of causal inference with machine learning.

In recent decades, diverse causal inference methods have emerged from various disciplinary backgrounds. Despite their proliferation, these methods often lack systematic

organization and cohesive theoretical guidance [24]. Theoretical frameworks such as potential outcomes [24] and structural causal models [26] have gradually emerged as leading theoretical systems in the field of causality, catalyzing new stages of development in causal theory research. Within these frameworks, methods for searching causal graph structures primarily encompass constraint-based methods, score-based methods, and hybrid methods [27]. Constraint-based methods like PC [28] and FCI [29] construct causal graphs by testing conditional independence relationships. Score-based methods such as GES [30], employing scoring functions like BDeu [31] and BIC [32], assess the quality of causal graphs and optimize these scoring functions using search algorithms. Hybrid methods such as the MMHC [19] and PC-MCMC algorithms [33] combine both constraint-based and score-based approaches, leveraging conditional independence tests and scoring functions to construct causal graphs. These methods find application in educational contexts, facilitating causal analysis of behaviors and learning outcomes [34]. They also enhance the performance and interpretability of cognitive diagnostics through causal discovery [35], thereby enabling personalized tutoring and resource recommendations in education. In summary, causal discovery technology plays an essential role in improving the sustainability of education. By deeply analyzing students' learning data and identifying and understanding the causal relationships in learning paths, more personalized education can be achieved, enhancing students' learning abilities and educational quality, thereby cultivating individuals who are more adaptable and innovative for the future society.

## 2.2. Literature Review of Graph Structure Search Based on Evolutionary Computational Methods

Causal relationships denote specific connections between objective events, where the occurrence, development, or change of one event (cause) influences another event (effect). These relationships are typically represented using causal graphs, with three common types being Directed Acyclic Graphs (DAGs), Partial Ancestral Graphs (PAGs), and Mixed Ancestral Graphs (MAGs) [36]. DAGs, the most widely used type, visually illustrate causal relationships by depicting causes pointing to effects.

The Learning Path Recognition (LPR) problem discussed in this paper addresses a graph structure search problem, transforming the prediction of learning paths into DAG forms to explore optimal learning sequences. Given the sparsity of directed edges representing knowledge components in the LPR problem, this paper employs a sparse adjacency matrix to represent learning path graph structures. The LPR problem is defined as a single-objective optimization problem, where the closer the obtained graph structure matches real learning paths during iterations, the higher the fitness values of individuals in the population.

The history of structural search using evolutionary algorithms dates back several decades, tracing its roots to pioneering studies [37,38]. Building on these early works, E. Real et al. [39] introduced a constrained evolutionary learning algorithm. In each iteration, this algorithm eliminates outdated structures and introduces newly generated ones, effectively searching for optimal structures across iterations. Evolutionary computing methods have proven effective in various real-world applications such as tourism route planning [40], program search [41], and ship traffic optimization [42]. By optimizing for specific tasks, evolutionary strategies also demonstrate remarkable performance in large-scale optimization tasks [43]. Recently, researchers have increasingly focused on leveraging evolutionary computing methods for structural search problems. Stanley [44], using genetic algorithms, developed an enhanced topological neural evolutionary network that evolves and optimizes network structures and weights [45] from basic unit structures. This method progressively approaches ideal performance through continuous evolution and selection of favorable traits. As the search space expands, strategies employing evolutionary algorithms for graph structure search are gaining popularity. These strategies excel not only in optimizing network weights but also in exploring and discovering new network structures. Salimans et al. [46] further validated the effectiveness of evolutionary algorithms in structural search, demonstrating their ability to achieve outstanding performance

comparable to traditional reinforcement learning methods. One significant advantage of evolutionary learning algorithms over reinforcement learning [47,48] lies in their ability to avoid the delayed reward problem. While reinforcement learning requires waiting for the entire sequence to end before evaluating actions for reward, evolutionary learning can select actions based on their fitness at each step, optimizing the network in real time. Additionally, Real et al. [49] proposed an evolutionary algorithm specifically for large-scale image classification tasks. Tested on CIFAR-10 and CIFAR-100 datasets, this algorithm demonstrated performance comparable to manually designed network models, indicating that evolutionary algorithms not only achieve high efficiency in handling complex tasks but also yield results comparable to those of human-designed models. During the iterative process, evolutionary algorithms can adaptively explore diverse learning paths for students. By collecting and analyzing students' learning data, an initial population is established where each individual represents a potential learning path. Subsequently, the algorithm evaluates each individual's performance using a fitness function, selects, recombines, and mutates individuals based on their performance to generate a new generation of learning path populations. Through multiple generations of iteration, the algorithm ultimately converges to a set of optimal learning paths. These optimal paths can be utilized to facilitate personalized learning, catering to the individual needs and abilities of each student. By employing evolutionary algorithms to identify learning paths, intelligent education systems can more effectively adapt to and meet the individualized needs of students, thereby enhancing educational quality, promoting equity and inclusivity, nurturing well-rounded talents, and ultimately achieving educational sustainability.

The application scope and potential of evolutionary algorithms are continually expanding. By optimizing and evolving network structures, researchers can more effectively tackle various complex tasks and challenges. This field of research not only provides new ideas and methods for designing and optimizing neural networks but also opens up new pathways for the development of artificial intelligence. With ongoing technological advancements and continuous methodological improvements, the role of evolutionary algorithms in network optimization is expected to become increasingly crucial, showcasing their unique advantages and value across diverse fields in the future.

### 3. Learning Path Recognition from Real-World Learning Data

#### 3.1. Characterizing Learning Paths with Graph Structures

This paper utilizes data from two primary sources: the Learning Path Recognition—Real World Datasets and the Learning Path Recognition—Generated Datasets, which were released as part of the NeurIPS 2022 competition “Causal Insights for Learning Paths in Education”. The real-world training data originate from Eedi’s online learning platform: <https://eedi.com> (accessed on 20 July 2022) and <https://diagnosticquestions.com> (accessed on 20 July 2022) and encompassing authentic data collected from students who completed tests and courses on the Eedi platform between 1 February and 3 August 2022. Eedi, an online education company registered in England and Wales, specializes in providing personalized learning experiences and AI-driven independent practice for students. Diagnostic questions administered to school students (approximately aged 11 to 16) assess their mastery of specific knowledge components. Each diagnostic question is multiple-choice, with four possible answers, of which only one is correct. Each lesson focuses on a specific knowledge point and includes instructional videos and self-marking questions [50]. This section aims to transform real-world data into a graph structure to analyze causal relationships between student performance levels under various learning sequences of knowledge components. The complete Real-World Dataset comprises 116 structures. To simplify solution complexity, NeurIPS also provides the Learning Path Recognition-Generated Datasets, each containing 50 structures. To ensure consistency with the dimensions of the generated datasets, the Learning Path Recognition—Real-World Datasets not only includes the complete Real-World Dataset of 116 structures but also two Real-World sub-datasets derived from the complete dataset to match the dimensions of the generated datasets. This facilitates testing the algorithms’ capabilities on datasets of similar

dimensions. It should be noted that although the data we used were from the 11–16 age group, the framework proposed in this study is age-independent. It is generalized and can be applied to any age group. Tables 1 and 2 outline the structure of the real-world data. In Table 1, QuizSessionId denotes the course ID, AnswerId indicates the student's answer ID, UserId represents the student ID, QuizId denotes the QuestionID, IsCorrect indicates the correctness of the student's answer, and AnswerValue (1–5) specifies which of the four options provided for the question the answer belongs to. In Table 2, CorrectAnswer (1–5) indicates the correct answer among the options, QuestionSequence (1–5) represents the sequence of questions, ConstructId denotes the knowledge point ID, and Type specifies the course type. The real dataset encompasses data from over 6400 students, including more than 65,000 test sessions, over 470,000 answers to diagnostic questions, and 37,000 course learning records [50].

**Table 1.** A total of 65,220 primary real-world answer data. Complete data can be found in (Gong et al., n.d.) [50].

QuizSessionId	AnswerId	UserId	QuizId	QuestionId	IsCorrect	AnswerValue
8	57	5	232,950	131,432	0	2
8	57	5	232,950	131,432	0	3
8	None	5	232,950	131,432	None	None
8	59	5	232,950	133,665	1	4
8	60	5	232,950	131,433	1	1

**Table 2.** Corresponding primary real-world lesson data.

CorrectAnswer	QuestionSequence	ConstructId	Type
4	2	433	Checkin
4	2	433	CheckinRetry
None	2	433	Lesson
4	2	433	Checkout
1	3	427	Checkin

This paper utilizes real-world data provided by the NeurIPS 2022 competition to model knowledge relationships through the construction of a graph structure. This graph structure takes the form of Directed Acyclic Graphs (DAGs), consisting of vertices and directed edges. The entirety of this graph structure forms a comprehensive knowledge graph, where entities and their relationships constitute fundamental elements for representing objective facts and building the knowledge graph. Within this framework, each vertex represents a knowledge point (ConstructId) that students are required to learn, functioning as an entity. By analyzing students' correct responses to questions related to various knowledge components (QuestionId), causal relationships between these points can be inferred. These causal relationships are depicted as sets of directed edges within the graph structure. For example, a directed edge (ij) from point i to point j indicates that learning knowledge point i influences learning knowledge point j, thereby reflecting a causal relationship between them.

### 3.2. The Method of the Study

Following the construction of the DAGs, this paper proceeds to derive students' learning paths based on real-world data. These learning paths are determined by tracing all knowledge components traversed from the starting point to the endpoint within the graph structure, thereby delineating complete learning trajectories. The Collaborative Structural Search Framework (CSSF) algorithm developed in this study aims to analyze these student learning paths derived from real-world data to identify optimal sequences for learning knowledge components. This approach to constructing knowledge graphs and optimizing paths based on graph structures holds significant implications not only for educational data analysis but also as a reference for analyzing causal relationships and optimizing pathways

in other domains. By conducting in-depth analyses of potential relationships within data, this framework enables enhanced comprehension and prediction of behavioral patterns within complex systems.

#### 4. Collaborative Structural Search Framework for LPR

##### 4.1. Collaborative Structural Search for LPR with Multiple Sub-Populations

To enhance the search efficiency of metaheuristic algorithms further, a novel approach based on multi-subpopulation evolutionary computing has been proposed. This method aims to augment fitness values through collaborative behaviors among multiple subpopulations [51–55]. Drawing on principles from swarm intelligence and collaborative search, the algorithm divides the search space into several subpopulations that work together to strike a balance between global exploration and local optimization. The framework of the Collaborative Structure Search Framework (CSSF) introduced in this study utilizes multiple subpopulations generated from the offspring population of Differential Evolution (DE). Each subpopulation is designed to fulfill distinct functions, evolving independently while also exchanging information with others. This organizational strategy is geared towards boosting population diversity, thereby enhancing search efficiency and global exploration capabilities. Tailored evolutionary strategies are employed in each subpopulation to align with the specific characteristics and search demands of the problem. The integration of winning subpopulations from the offspring introduces a positive feedback mechanism. This involves comparing these winners with individuals in the current population, analyzing increases in the number of edges within the causal structure, and incorporating a predefined positive feedback probability (PF) to introduce directed edges to population individuals. Conversely, the incorporation of elimination subpopulations from the offspring introduces a negative feedback mechanism. This process entails comparing elimination subpopulations with individuals in the current population, analyzing increases in the number of edges within the causal structure, and employing a predefined negative feedback probability (NF) to remove directed edges from population individuals.

Subsequently, individuals with low fitness from the current population are eliminated, while those with high fitness from the offspring population undergo crossover operations. Following this, the current population segregates into winning and elimination populations. Winning populations, characterized by superior fitness, are retained for the subsequent iteration, while elimination populations are discarded. This iterative process continues until a specified number of iterations is reached, aiming to achieve the lowest loss value and identify the best individual. In this context, an individual's loss is defined as the difference between 1 and its fitness value  $F$ , where both  $F$  and loss are normalized values between 0 and 1. Here, loss indicates the degree of discrepancy between the individual and the actual graph structure, while FFF indicates the degree of similarity between them. For instance,  $F$  may be derived using the F1-Score calculation method, thereby characterizing the loss as:

$$\left\{ \begin{array}{l} \text{loss} = 1 - F(I) \\ \text{s.t. } F(I) = \text{F1} - \text{Score}_I \\ \quad \text{precision}_I = \frac{\text{TP}_I}{\text{TP}_I + \text{FP}_I} \\ \quad \text{recall}_I = \frac{\text{TP}_I}{\text{TP}_I + \text{FN}_I} \\ \quad \text{F1} - \text{Score}_I = 2 * \frac{\text{precision}_I * \text{recall}_I}{\text{precision}_I + \text{recall}_I} \end{array} \right. \quad (1)$$

The F1-Score serves as a critical metric in classification tasks, quantifying the harmonic mean of precision and recall on a scale from 0 to 1. A score of 1 indicates perfect classification, while 0 signifies complete misclassification. In this context,  $TP$  (True Positive) represents correctly predicted positive samples,  $FP$  (False Positive) indicates samples incorrectly predicted as positive,  $TN$  (True Negative) denotes correctly predicted negative samples, and  $FN$  (False Negative) reflects samples incorrectly predicted as negative. Here, the subscript  $I$  denotes an individual, and  $F(I)$  signifies the process of evaluating the fitness value for the individual  $I$ .

Given the diverse nature of problems, researchers have the flexibility to employ various evaluation functions to compute  $F$ , tailored to specific problem requirements. This adaptability enhances the extensibility of the Collaborative Structure Search Framework (CSSF) discussed in this study.

The design philosophy of CSSF emphasizes both generality and scalability, facilitating its application to a wide array of complex graph structure optimization problems. Researchers can select appropriate evaluation functions that best align with the problem's characteristics. For instance, precision may be prioritized in scenarios where minimizing false positives is crucial, whereas higher recall might be necessary to maximize true positive identifications in other contexts. By adjusting the evaluation function accordingly, CSSF can effectively cater to diverse optimization objectives, thereby yielding optimal solutions across varied application scenarios. This inherent flexibility not only enhances the practical applicability of CSSF but also allows for continual algorithmic enhancements and refinements. Developers can iteratively fine-tune the evaluation function to improve the framework's performance and efficacy in addressing evolving challenges in graph structure optimization. Consequently, CSSF emerges not only as a robust solution for current applications but also as a versatile tool capable of adapting to future technological advancements and demands in the field.

In summary, the CSSF framework stands out for its capability to deliver high-performance outcomes across complex graph structure problems, underscored by its adaptive design and potential for continual improvement.

#### 4.2. The Structure of the Proposed Framework

This paper addresses the challenge of identifying authentic learning paths within the educational domain. By transforming causal relationships from real-world learning datasets into graph-structured data, the task of identifying learning paths becomes increasingly intricate and demanding as students master more knowledge components [56]. Consequently, navigating through all potential graph structures that represent learning paths constitutes a typical NP-hard problem. To address this formidable computational complexity in graph structure search, researchers have explored various heuristic search algorithms aimed at discovering the optimal network structure within the solution space.

In this context, Gao et al. [57] proposed a graph structure search algorithm based on ant colony optimization. This approach constructs an undirected graph framework and utilizes the ant colony optimization method to conduct a stochastic search within the solution space, ultimately generating a directed acyclic graph (DAG). Conversely, Yang et al. [58] introduced a graph structure search method based on bacterial foraging optimization, characterized by the integration of mechanisms such as chemotaxis, reproduction, elimination, and dispersal, and employing the K2 scoring function to evaluate the inferred graph structure.

If we conceptualize the graph structure search problem as a combinatorial optimization challenge, it can be mathematically modeled as follows:

$$\begin{cases} \text{MaxScore}(G|D), G \in S \\ \text{s.t } G \in \Gamma \end{cases} \quad (2)$$

Or:

$$\begin{cases} \text{MinScore}(G|D), G \in S \\ \text{s.t } G \in \Gamma \end{cases} \quad (3)$$

In the context of this problem,  $\text{Score}(G|D)$  represents the scoring function value of the predicted graph structure  $G$  given the sample dataset  $D$ . The set  $S$  encompasses all candidate structures in the search space, for which  $G \in \Gamma$  indicates that the predicted graph structure  $G$  satisfies the specified constraint conditions  $\Gamma$ . In practical applications where no additional constraints are specified,  $G \in \Gamma$  implies that the candidate structure  $G$  must adhere to the acyclic property. This paper proposes a multi-subgroup collaborative

evolutionary algorithm framework named CFFS (Collaborative Subgroup Structure Search) and leverages the CSSF (Collaborative Subgroup Structure Framework) algorithm for path identification and optimization. The CSSF algorithm framework facilitates collaborative graph structure search among multiple subgroups, enabling efficient discovery of optimal learning paths.

In the proposed algorithm framework, each graph structure is represented as a one-dimensional matrix. If there are  $n$  knowledge components in the graph structure, denoted as  $X_1, X_2, \dots, X_n$ , the parent set of each knowledge point  $X_i$  is denoted as  $G = \{GP(X_1, X_2, \dots, X_n)\}$ . The parent sets of all knowledge components collectively form a complete learning path in the graph structure. CSSF introduces an environmental pressure parameter AP as a hyperparameter, representing the proportion of the elite population within the entire population, ranging from 0 to 1. The parameter maxFE denotes the maximum allowed function evaluations, where FE increases with each population iteration. The algorithm terminates when FE reaches  $\geq$  maxFE. During each iteration, each offspring individual undergoes individual optimization operations based on positive and negative feedback mechanisms to align the distribution of decision variables with the real learning path more effectively. Subsequently, the optimized offspring population merges with the parent population. The combined population is sorted based on fitness values and divided from highest to lowest into elite and eliminated subpopulations. The eliminated subpopulation is then discarded based on its fitness level and replenished using the differentiation transfer strategy. The elite offspring subpopulation merges with the current population to form the parent population for the subsequent iteration.

Detailed strategies introduced will be further elaborated in subsequent sections of this paper. Each time the objective value is calculated, the evaluation count FE is incremented by one. The algorithm ceases operation once the predefined maximum evaluation count maxFE is reached. Algorithm 1 outlines the proposed structure search framework in detail.

The execution process of the proposed Collaborative Subgroup Structure Framework (CSSF) algorithm is detailed in Algorithm 1. The core workflow of this algorithm unfolds as follows: initially, the population is randomly initialized, and the function evaluation count (FE) is set to zero (Lines 1–2 of Algorithm 1). Subsequently, the algorithm enters an iterative evolution process. In each iteration, a new generation of offspring populations is generated by applying the Differential Evolution (DE) algorithm to the current population. The individuals in the offspring population are then sorted according to a predefined optimization direction (Lines 4–5 of Algorithm 1).

Following the sorting of the offspring population, it is divided into superior and inferior sub-populations. These sub-populations undergo edge addition and deletion operations based on positive feedback probability (PF) and negative feedback probability (NF), respectively. The effectiveness of these mechanisms is assessed by comparing the number of new directed edges added or removed by the superior and inferior sub-populations relative to the current population (Lines 6–11 of Algorithm 1).

After executing the edge addition and deletion operations based on feedback mechanisms, the offspring population and the current population are sorted by fitness. Less fit individuals from the current population are eliminated, while more fit individuals from the offspring population are retained (Lines 12–15 of Algorithm 1). Subsequently, under the influence of ambient pressure (AP), the current population is categorized into three parts: the least fit individuals are discarded, the most fit individuals proceed directly to the next iteration, and the remaining individuals undergo a diversification transfer strategy to maintain population diversity for the next iteration (Lines 16–18 of Algorithm 1).

The while loop encompassing Lines 3–19 in Algorithm 1 iterates until a termination condition is met. Upon reaching the maximum function evaluation count (maxFE), the algorithm ceases iteration and outputs the optimal individual (Lines 20–21 of Algorithm 1).

**Algorithm 1:** Collaborative Structural Search Framework for LPR

---

```

Input    : Population optimization direction: optimization_direction, Environmental
           pressure: ambient_pressure
Output   : The best individual Ibest and the corresponding system loss
1: Initialization: Initialize the population randomly
2: FE ← 0
3: while FE < maxFE do
4:     The parent population Pop generates the offPop of the child population by the
       DE algorithm
5:     OffPop ← sort([ OffPop, optimization_direction])
6:     //Population fitness ranking
7:     //Establish a positive and negative feedback mechanism
8:     for i = ambient_pressure * PopSize: PopSize or i: ambient_pressure * PopSize - i do
9:         | NewEdge ← Offspring(1,i).dec - Population(1,i).dec
10:        | count_ones ← count_ones + NewEdge
11:        | //Record the number of new edges
12:    end
13:    count_ones* ← NF or PF
14:    for i = ambient_pressure * PopSize: PopSize or i = 1: ambient_pressure * PopSize - 1 do
15:        | OffPop(1,i).dec ← 1; or OffPop(1,i).dec ← 0;
16:    end
17:    replace ← FitnessSingle(Pop) - FitnessSingle(OffPop) > 0
       //The fitness of the offspring population was compared with the current population
18:    Pop(replace) ← OffPop(replace);
       Pop(1 : ambient_pressure * PopSize) ← Pop((PopSize - ambient_pressure * PopSize +
19: 1): PopSize)
       //Perform a differentiation transfer strategy
20:    FE ← FE + PopSize
       //During the sorting process, the offspring population requested the evaluation of the
       PopSize secondary function
21: end
22: Find out the best individual Ibest and the corresponding system loss generated during the
   iteration process.
23: return Ibest and loss

```

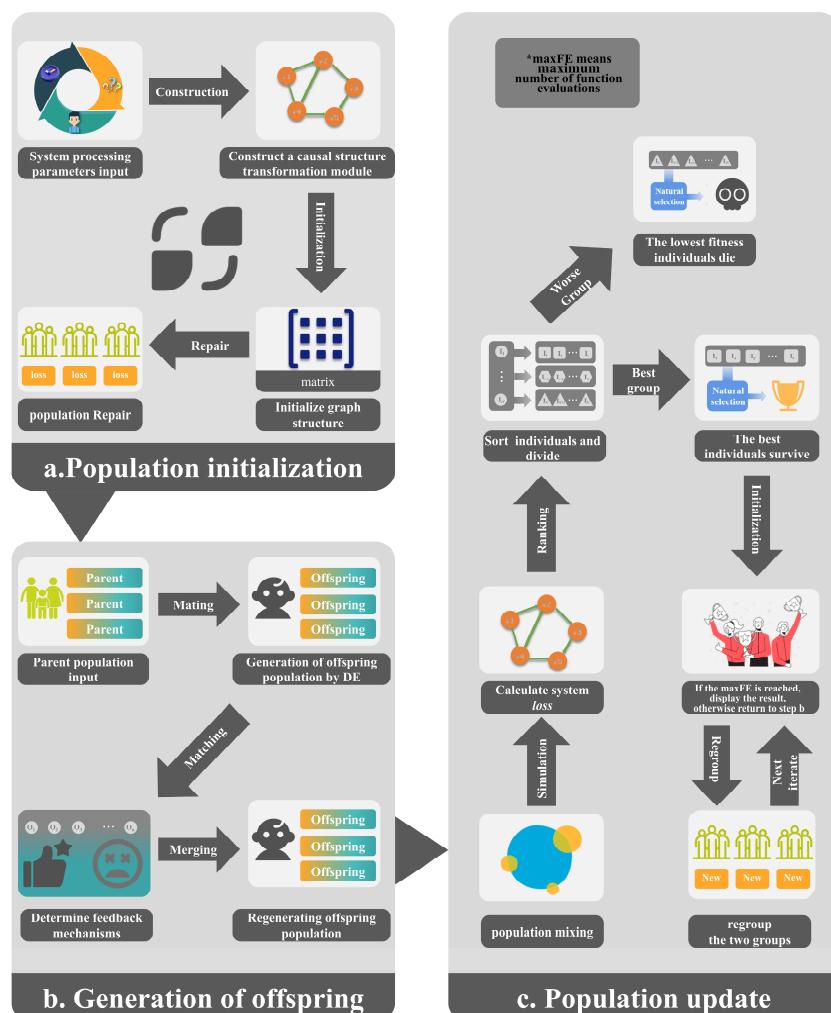
---

During the initialization phase, random generation of the initial population ensures exploration across a diverse range of points in the solution space, thereby mitigating the risk of converging to local optima. Employing the DE algorithm in each iteration simulates biological evolution's variation and selection processes, progressively enhancing population fitness.

The segmentation of the offspring population into superior and inferior sub-populations, combined with positive and negative feedback mechanisms, represents a significant innovation of the CSSF algorithm. The positive feedback mechanism enriches effective causal relationships by augmenting directed edges in the superior sub-population, thereby enhancing overall fitness. Conversely, the negative feedback mechanism reduces unnecessary or conflicting relationships by reducing directed edges in the inferior sub-population, further refining the population structure. Introducing fitness-based sorting and ambient pressure ensures continuous enhancement of population fitness while preserving diversity. Direct advancement of highly fit individuals to the next iteration facilitates rapid propagation of beneficial traits, while the diversification transfer strategy preserves individuals to sustain population diversity, guarding against premature convergence to local optima.

Ultimately, when the predetermined function evaluation count (maxFE) is reached, the algorithm halts and identifies the optimal solution within computational constraints, underscoring its efficiency and reliability. Through iterative optimization and adaptive feedback mechanisms, the CSSF algorithm excels in complex learning path identification and optimization tasks, particularly with extensive real-world datasets, demonstrating robust performance and reliability.

Figure 1 illustrates the flowchart of the Collaborative Subgroup Structure Framework (CSSF) algorithm, detailing its implementation process as proposed in this study. The algorithm's core steps include individual optimization and a differentiation transfer strategy, consisting of three main components: generating offspring populations, employing a positive–negative feedback mechanism, and selecting crossovers. During algorithm execution, the first step involves checking whether the function evaluation count (FE) has reached its maximum value. If not, the algorithm initializes the population. During this initialization phase, randomly generated populations may contain erroneous cycles, necessitating individual repair to eliminate cycles from paths and ensure population validity. Subsequently, the parent population generates offspring populations, which are then partitioned into winning and elimination subgroups. Through the positive–negative feedback mechanism, the offspring populations undergo further optimization. The resultant offspring individuals proceed to the individual optimization phase to enhance their fitness and overall performance. Following individual optimization, the optimized offspring population merges with the current population and undergoes fitness-based sorting. The sorted population is segregated according to the specified optimization direction, leading to the division into winning and elimination subgroups. The winning subgroup is retained as the parent population for the subsequent iteration, while the elimination subgroup is discarded. This iterative process incrementally enhances population fitness with each iteration, approaching the optimal solution.



**Figure 1.** Material processing flow chart. By learning from continuously updated educational data, the proposed algorithm is promoted to sustainably evolve and develop in education. In the figure, \*maxFE represents the maximum number of times the algorithm can be run.

This method ensures continuous population improvement and optimization across generations through individual optimization, a differentiation transfer strategy, and a positive–negative feedback mechanism. During the generation of offspring populations, fitness-based sorting and selection retain superior individuals while eliminating weaker ones, thereby bolstering overall population fitness. Ultimately, the algorithm effectively explores the solution space and consistently enhances solution quality.

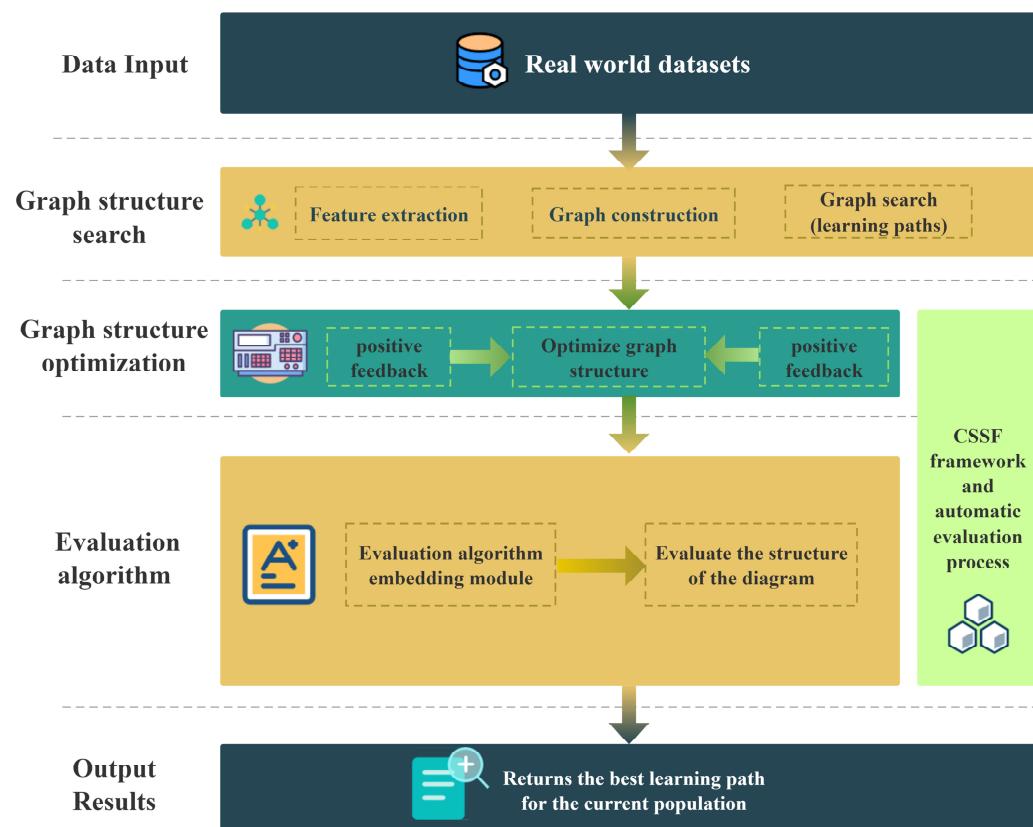
#### 4.3. Identify Effective Learning Paths with CSSF

The CSSF algorithm proposed in this paper is a versatile algorithmic framework specifically designed for optimizing and analyzing problems involving graph structures. Its design emphasizes high flexibility and adaptability by seamlessly integrating with any evaluation algorithm. This integration allows for the synergistic evolution and optimization of graph structures. The overall data flow of the CSSF algorithm framework can be delineated into several key steps. Initially, the algorithm analyzes input graph structure data and transforms it into a Directed Acyclic Graph (DAG). This preprocessing step ensures computational efficiency and simplifies subsequent processing. Building upon the repaired DAG, the algorithm generates a population of individuals that represent potential solutions. These individuals serve as starting points for the algorithmic search and optimization processes. Simultaneously, the CSSF algorithm incorporates a dual feedback mechanism—positive and negative—which can be coupled with various evaluation algorithms. These evaluation algorithms provide real-time feedback based on the performance of the current population of individuals during each iteration. This feedback guides adjustments to population parameters and optimization of learning paths. Specifically, the evaluation algorithm assesses the quality of the current population of individuals in each iteration, generating feedback scores used to direct the evolution of the population and optimize learning paths. It evaluates the generated graph structures, analyzing their effectiveness and providing evaluative scores. Based on these evaluation results, adjustments or optimizations are made to the graph structures and corresponding learning paths. CSSF integrates graph structure search and automated evaluation processes into a unified framework. Within this framework, results from graph structure searches are automatically evaluated by embedded evaluation algorithms, which subsequently inform optimization strategies based on evaluation outcomes. The orthogonality of the evaluation algorithm is a pivotal feature of the CSSF algorithm. This characteristic enables seamless integration with the CSSF framework, accommodating diverse evaluation methods ranging from rule-based to machine learning-based approaches. Such versatility significantly enhances the generality and adaptability of the CSSF algorithm, making it applicable to a wide array of complex graph structure optimization problems.

The design of the CSSF algorithm framework offers significant advantages. By transforming input data into DAG, CSSF simplifies computational complexity and enhances processing efficiency. The incorporation of a dual feedback mechanism—positive and negative—enables CSSF to dynamically adjust population individuals, thereby enhancing adaptability and optimization capabilities. Integrating graph structure search with automated evaluation processes within a unified framework facilitates efficient interaction and collaborative optimization. The orthogonal design of the evaluation algorithm allows CSSF to accommodate various evaluation methods, making it widely applicable to different types of graph structure optimization problems.

In summary, the CSSF algorithm framework proposed in this paper is a powerful and versatile tool. Through its orthogonal integration with any evaluation algorithm, CSSF achieves optimized analysis of graph structure problems. Its efficient data flow processing, flexible feedback mechanisms, and integration of automated evaluation and optimization underscore its excellent performance in solving complex optimization problems. With this design, CSSF demonstrates robust performance and broad applicability across diverse real-world applications, providing an effective solution for optimizing graph structure problems.

Figure 2 illustrates the operational flow of the CSSF algorithm framework. Initially, real-world data are input into the CSSF framework. The algorithm then conducts a graph structure search on the learning paths and initializes the predicted learning path. The CSSF algorithm integrates positive and negative dual feedback mechanisms with differentiation transfer strategies to optimize population individuals, thereby efficiently identifying the optimal predicted learning path. Simultaneously, during the evaluation of graph structures, any evaluation algorithm can be seamlessly embedded into the CSSF algorithm framework, significantly enhancing its versatility and adaptability. This process underscores the CSSF algorithm's robust adaptability and flexibility in various practical applications. Real-world data input is leveraged to ensure the practical significance and accuracy of predicting learning paths. Throughout the graph structure search, the CSSF algorithm ensures the effectiveness of the initial learning path. The positive and negative dual feedback mechanism enables dynamic adjustments and optimizations, swiftly approaching the optimal path. The differentiation transfer strategy further enhances population diversity and exploration capabilities, mitigating the risk of local optima. Moreover, the plug-and-play feature of the evaluation algorithm allows the CSSF framework to flexibly adapt to different evaluation criteria and requirements, whether based on performance evaluation or other customized methods. This design enhances the algorithm's versatility and expands its application scope across diverse domains.



**Figure 2.** Framework and data flow of the CSSF algorithm.

## 5. Simulation Results and Analysis

### 5.1. Experimental Settings

The simulation experiments in this study utilized the Learning Path Recognition (LPR) dataset as the basis. To ensure experimental fairness, all algorithms employed “loss” as the evaluation metric. The experiments were conducted on a computing device equipped with an Intel i5-13600KF@3.50 GHz dual-core processor, 32 GB of operating memory, and an NVIDIA RTX 4070 graphics card. The software environment comprised Matlab 2022b and the PlatEMO version 4.5 (Platform for Evolutionary Multi-Objective Optimization) [59].

PlatEMO Developed by BIMK (Institute of Bioinspired Intelligence and Mining Knowledge) of Anhui University. Visualization tasks were performed using tools available on chiplot.online. It is noteworthy that, apart from the proposed CSSF algorithm in this paper, all comparison algorithms utilized parameter combinations either as originally specified in their respective papers or default parameters provided by the PlatEMO platform. Hyperparameters specific to the comparison algorithms are detailed in Table 3. Among the comparison algorithms, the MSEA algorithm proposed by Y. Tian et al. [20] operates as a constraint-based method that partitions the optimization process into multiple stages to maintain diverse populations. The Multi-Form Optimization Framework Algorithm (MFOSPEA2) introduced by R. Jiao et al. [60] significantly accelerates the discovery of optimal solutions by integrating various search forms and strategies within a multi-form optimization framework. This approach enhances adaptability and solution efficiency in complex optimization scenarios, enabling rapid identification of high-quality solutions. The Golden Eagle Optimization Algorithm (GEO) [21] mimics the multi-stage hunting behavior of golden eagles, aiming to capture the best prey in feasible areas swiftly, thereby achieving superior fitness values in global optimization. The Harris Hawk Optimization Algorithm based on the Elite Evolution Strategy (EESHHO) [61] improves upon traditional evolutionary strategies by simulating the dynamic hunting behavior of Harris hawks and integrating an elite evolution mechanism. This strategy accelerates the optimization process by prioritizing individuals with higher fitness for reproduction and mutation, thus enhancing solution quality and balancing global and local searches. Specific hyperparameters for the CSSF algorithm proposed in this paper are outlined in the table below. The experimental setup involved using LPR series test problems on the PlatEMO platform version 4.5. Readers intending to utilize the provided source code or data from this study are advised to install the PlatEMO platform and its associated components. The source code and data files can be downloaded from the GitHub project homepage of this study <https://github.com/YuanHao-CS/CSSF>, accessed on 20 July 2022 [62]. The project accessed on 19 June 2024. In the experimental process, the population size for all algorithms was fixed at 100 to ensure consistency, and the maximum evaluation limit for all algorithms was set to 10,000.

**Table 3.** Hyper-parameter settings of comparison algorithms.

Algorithm	Parameter
MFOSPEA2 [60]	$Initial_{max} = 1, Initial_{min} = 0$
GEO [21]	$AP_{min} = 0.5, AP_{max} = 2.0, CP_{min} = 1.0, CP_{max} = 0.5$
EESHHO [61]	$Ub = 1, Lb = 0$
MSEA [20]	$F_{max} = 1, F_{min} = 0$
CSSF	$PF = 0.6, NF = 0.4, AP = 0.4$

This paper comprises five distinct sets of experiments designed to thoroughly evaluate the effectiveness and performance of the CSSF algorithm in path recognition. Each experiment group is detailed as follows:

In Section 5.2, we conducted comparison experiments using generated datasets. These experiments aimed to validate the effectiveness of the CSSF algorithm in path recognition by comparing its performance with other algorithms (LPR-GD1 to LPR-GD5) across various path recognition tasks. The experimental results unequivocally demonstrated the CSSF algorithm's significant outperformance of other algorithms on these generated datasets, establishing its superiority.

In Section 5.3, tests were conducted using real-world datasets, specifically LPR-RWD1 to LPR-RWD3. These experiments aimed to verify the CSSF algorithm's performance in practical applications. Real-world datasets typically contain more noise and uncertainty, making them more challenging and reflective of practical scenarios. The results showcased the CSSF algorithm's excellent performance on these real-world datasets, exhibiting su-

rior recognition accuracy and computational efficiency compared to other algorithms, thus confirming its feasibility and effectiveness in real-world scenarios. Section 5.4 involved further Friedman rank tests. These experiments not only comprehensively compared the CSSF algorithm with other algorithms but also validated its sustained superiority, particularly highlighted in its performance with real-world datasets. This result further underscores the robustness and reliability of the CSSF algorithm in path recognition tasks.

In Section 5.5, ablation experiments were conducted to explore the underlying reasons for the CSSF algorithm's outstanding performance in path recognition problems. Specifically, we compared experiments with and without the dual-feedback mechanism. The results showed that the CSSF algorithm with the dual-feedback mechanism significantly outperformed the version without it across multiple performance metrics, thereby confirming the positive impact of this mechanism on algorithm performance. This finding solidifies the effectiveness and innovation of the CSSF algorithm in path recognition. Section 5.6 focused on convergence experiments to analyze the CSSF algorithm's performance across different problems, illustrated through evolutionary curves and box plots. The results demonstrated the CSSF algorithm's excellent convergence on all problems, exhibiting minimal influence from randomness. This indicates that the CSSF algorithm not only quickly finds optimal solutions for various path recognition problems but also maintains stable performance that is less susceptible to dataset variability, validating its stability and reliability in practical applications.

Through these rigorous experimental validations, we have obtained a comprehensive understanding of the superiority and effectiveness of the CSSF algorithm in path recognition. These findings not only provide compelling evidence for further research and optimization of the CSSF algorithm but also establish a robust theoretical and experimental foundation for its promotion and implementation in practical applications. Future research can build upon these results to explore the CSSF algorithm's potential in broader application scenarios and further optimize its performance to address more complex path recognition challenges.

### 5.2. Test Experiments on Generated Datasets

In this experimental stage, the algorithm proposed in this paper was evaluated against several other algorithms using datasets generated from LPR-GD1-5. Each algorithm underwent a minimum of three runs per problem across the five datasets to mitigate the impact of randomness on the experimental outcomes. Synthetic datasets with predefined causal structures were employed to facilitate a more intuitive comparison of algorithm performances. The findings of this comparative analysis are consolidated and presented in Table 4.

**Table 4.** The average loss and standard deviation of the five algorithms in the LPR-GD series of problems. The optimal algorithm in each scenario has been highlighted with a gray background to facilitate the comparison of effects between algorithms.

Problem	D	MSEA [20]	GEO [21]	EESHHO [61]	MFOSPEA2 [60]	CSSF
LPR-GD1	1225	$4.5028 \times 10^{-1}$ ( $1.57 \times 10^{-3}$ ) [-]	$3.3908 \times 10^{-1}$ ( $8.61 \times 10^{-4}$ ) [-]	$3.2998 \times 10^{-1}$ ( $1.46 \times 10^{-2}$ ) [-]	$4.4859 \times 10^{-1}$ ( $6.59 \times 10^{-3}$ ) [-]	$2.7881 \times 10^{-1}$ ( $1.16 \times 10^{-2}$ )
LPR-GD2	1225	$4.3931 \times 10^{-1}$ ( $6.38 \times 10^{-3}$ ) [-]	$3.1985 \times 10^{-1}$ ( $7.88 \times 10^{-4}$ ) [-]	$3.0356 \times 10^{-1}$ ( $2.76 \times 10^{-3}$ ) [-]	$4.4236 \times 10^{-1}$ ( $2.87 \times 10^{-3}$ ) [-]	$2.6862 \times 10^{-1}$ ( $1.19 \times 10^{-2}$ )
LPR-GD3	1225	$4.5636 \times 10^{-1}$ ( $8.79 \times 10^{-3}$ ) [-]	$3.5202 \times 10^{-1}$ ( $5.45 \times 10^{-4}$ ) [-]	$3.4858 \times 10^{-1}$ ( $3.93 \times 10^{-3}$ ) [-]	$4.5749 \times 10^{-1}$ ( $4.65 \times 10^{-3}$ ) [-]	$2.8766 \times 10^{-1}$ ( $3.15 \times 10^{-3}$ )
LPR-GD4	1225	$4.5344 \times 10^{-1}$ ( $8.70 \times 10^{-4}$ ) [-]	$3.4933 \times 10^{-1}$ ( $5.63 \times 10^{-4}$ ) [-]	$3.3493 \times 10^{-1}$ ( $1.27 \times 10^{-2}$ ) [-]	$4.5336 \times 10^{-1}$ ( $6.86 \times 10^{-3}$ ) [-]	$2.8556 \times 10^{-1}$ ( $3.81 \times 10^{-3}$ )
LPR-GD5	1225	$4.4734 \times 10^{-1}$ ( $5.53 \times 10^{-3}$ ) [-]	$3.3507 \times 10^{-1}$ ( $7.74 \times 10^{-4}$ ) [-]	$3.2426 \times 10^{-1}$ ( $1.44 \times 10^{-2}$ ) [-]	$4.5020 \times 10^{-1}$ ( $2.98 \times 10^{-3}$ ) [-]	$2.7334 \times 10^{-1}$ ( $8.04 \times 10^{-3}$ )
+/-/-	-	0/0/5	0/0/5	0/0/5	0/0/5	-

In Table 4, we investigated five distinct test problems denoted as LPR-GD1 through LPR-GD5. In this table, we use the symbols “+”, “=”, and “−” to indicate the number of problems where a comparison algorithm performs better than, equal to, or worse than the proposed algorithm, respectively. Each problem encompasses a simulated adjacency matrix of learning paths, comprising 50 knowledge components with a dimensionality ( $D$ ) of 1225. Initially, a horizontal comparison of various algorithms was conducted. The values presented in the table denote the average loss values obtained from multiple runs of each algorithm on the corresponding test problems, where a smaller loss indicates superior algorithm performance.

From the table, it is evident that the MSEA and MFOSOE2 algorithms exhibit similar effectiveness, showing comparable results. Conversely, the GEO and EESHHO algorithms outperform the former two in solving this optimization problem, particularly the EESHHO algorithm. Notably, the CSSF algorithm proposed in this study demonstrates the most favorable performance across all test problems. This underscores that the multi-subpopulation collaboration strategy devised in this research significantly enhances the resolution of the LPR series problems.

It is worth mentioning that the MSEA and MFOSOE2 algorithms show similar performance on LPR-GD3 and LPR-GD4 and perform least effectively on the LPR-GD1 problem. In contrast, both the EESHHO and CSSF algorithms consistently achieve the best performance across all problems. Based on the insights gleaned from the table, it can be concluded that the optimization complexity of the LPR-GD2 problem is notably lower compared to the other problems. This further corroborates the varying performance of different algorithms across different problem instances, as well as the superiority of the CSSF algorithm in addressing diverse LPR problems. These findings contribute essential insights toward a deeper comprehension of the characteristics of LPR problems and the efficacy of optimization algorithms.

### 5.3. Real-World Datasets Comparison Experiment

In this experimental phase, the CSSF algorithm proposed in this paper has been benchmarked against several other algorithms across three distinct problems, namely LPR-RWD1 to LPR-RWD3, which utilize real datasets. Each algorithm underwent a minimum of three runs per test problem, and the findings of this comparative study have been consolidated in Table 5.

**Table 5.** The average loss and standard deviation of the five algorithms in the LPR-RDW series of problems. The optimal algorithm in each scenario has been highlighted with a gray background to facilitate the comparison of effects between algorithms.

Problem	$D$	MSEA [20]	GEO [21]	EESHHO [61]	MFOSPEA2 [60]	CSSF
LPR-RWD	6670	$4.8361 \times 10^{-1}$ ( $1.78 \times 10^{-3}$ ) [-]	$3.4382 \times 10^{-1}$ ( $3.95 \times 10^{-5}$ ) [-]	$3.4306 \times 10^{-1}$ ( $2.76 \times 10^{-4}$ ) [≈]	$4.8325 \times 10^{-1}$ ( $2.92 \times 10^{-3}$ ) [-]	$3.3844 \times 10^{-1}$ ( $4.45 \times 10^{-3}$ )
LPR-RWD1	1225	$4.1256 \times 10^{-1}$ ( $4.36 \times 10^{-3}$ ) [-]	$2.7243 \times 10^{-1}$ ( $2.06 \times 10^{-3}$ ) [-]	$2.7075 \times 10^{-1}$ ( $1.59 \times 10^{-3}$ ) [-]	$4.2106 \times 10^{-1}$ ( $2.68 \times 10^{-3}$ ) [-]	$2.3111 \times 10^{-1}$ ( $4.25 \times 10^{-3}$ )
LPR-RWD2	1225	$4.4332 \times 10^{-1}$ ( $8.90 \times 10^{-3}$ ) [-]	$3.3420 \times 10^{-1}$ ( $2.15 \times 10^{-4}$ ) [-]	$3.2095 \times 10^{-1}$ ( $1.12 \times 10^{-2}$ ) [-]	$4.4727 \times 10^{-1}$ ( $2.27 \times 10^{-3}$ ) [-]	$2.7595 \times 10^{-1}$ ( $3.87 \times 10^{-3}$ )
+/-/-	-	0/0/3	0/0/3	0/1/2	0/0/3	-

In Table 5, we explore three distinct test problems: LPR-RWD1 to LPR-RWD3, which are based on real dataset scenarios. Specifically, LPR-RWD1 and LPR-RWD2 extract adjacency matrix data for nodes 1 to 50 and 51 to 100, respectively, while LPR-RWD encompasses the complete adjacency matrix of 116 nodes. By analyzing these subsets of data, our objective is to evaluate whether the CSSF algorithm proposed in this study demonstrates performance advantages across different dimensions of real datasets. Here,  $D$  represents the dimensionality of the respective problem. The values presented for each algorithm in Table 5 represent the average loss after multiple runs on the corresponding test problem,

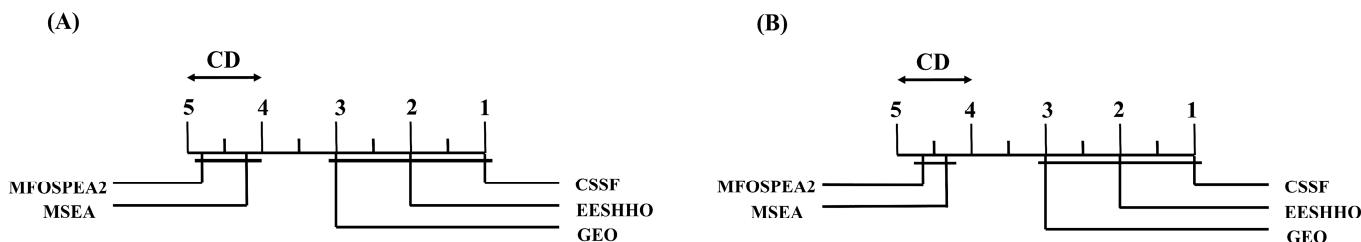
where a smaller loss signifies superior algorithm performance. Similar to the findings from experiments with synthetic datasets, MSEA and MFOSPEA2 demonstrate comparable performance, albeit generally less effective across various test problems compared to other benchmark algorithms. Conversely, GEO and EESHHO perform similarly to the CSSF algorithm when addressing the complete adjacency matrix problem in LPR-RWD. However, they notably underperform the CSSF algorithm in LPR-RWD1 and LPR-RWD2. Overall, the CSSF algorithm consistently exhibits the best performance across all problems, validating its efficacy in handling real datasets. Further analysis reveals that as problem dimensionality increases, the complexity associated with learning path recognition and exploration also rises gradually. This observation underscores the intricate nature of real datasets and highlights the challenges algorithms face when tackling higher-dimensional problems. In a vertical comparison, GEO and EESHHO demonstrate similar performance on LPR-RWD and LPR-RWD1. However, across the LPR-RWD series problems, irrespective of problem dimension, EESHHO and DE-TS consistently achieve the best performance, consistent with earlier experimental outcomes. Additionally, the table underscores that the optimization difficulty of LPR-RWD problems surpasses that of the other two problems, with LPR-RWD2 presenting a slightly greater challenge than LPR-RWD1. These insights gleaned from the experiments provide valuable clues for gaining a deeper understanding of algorithm performance characteristics when confronted with real datasets.

#### 5.4. Friedman Test Ranking Experiment

Based on the two aforementioned comparative experiments, further analysis was conducted using the Friedman test to evaluate the systematic loss values obtained by all algorithms. The Friedman test is a statistical method that ranks the performance of all comparison algorithms across all test problems, as depicted in Table 6. In Figure 3A,B's CD diagrams, lower rankings indicate greater efficiency of the algorithms compared to others. From the experiments, it is evident that the CSSF algorithm proposed in this paper consistently achieves the highest rank, demonstrating its superiority over the comparative algorithms across both real-world and synthetic datasets. Among the other algorithms compared, the EESHHO and GEO algorithms perform the next best, with rankings of 2.0 and 3.0, respectively, in both real-world and synthetic datasets. The difference in rank between the MFOSPEA2 and MSEA algorithms is marginal, with the MSEA algorithm exhibiting slightly better performance than MFOSPEA2.

**Table 6.** Friedman test results on real-world datasets and synthetic datasets.

Algorithm	CSSF	EESHHO	GEO	MSEA	MFOSPEA2
Real-world Rank	1.00	2.00	3.00	4.30	4.70
Synthetic Rank	1.00	2.00	3.00	4.20	4.80
Standard deviation	0.00	0.00	0.00	0.07	0.07



**Figure 3.** Results of Friedman tests on the fitness values of five algorithms across different datasets. (A) Friedman test results on generated datasets for the five algorithms, (B) Friedman test results on real-world datasets for the five algorithms. Here, the CD represents significant differences. Subplots (A,B) illustrate the Friedman test results for generated and real-world datasets, respectively.

The results of the Friedman test conducted on the real-world and synthetic dataset problems closely align with previous experimental findings, further confirming the superior performance of the CSSF algorithm. This underscores that the CSSF algorithm proposed in this study excels in addressing both real-world and simulated optimization problems.

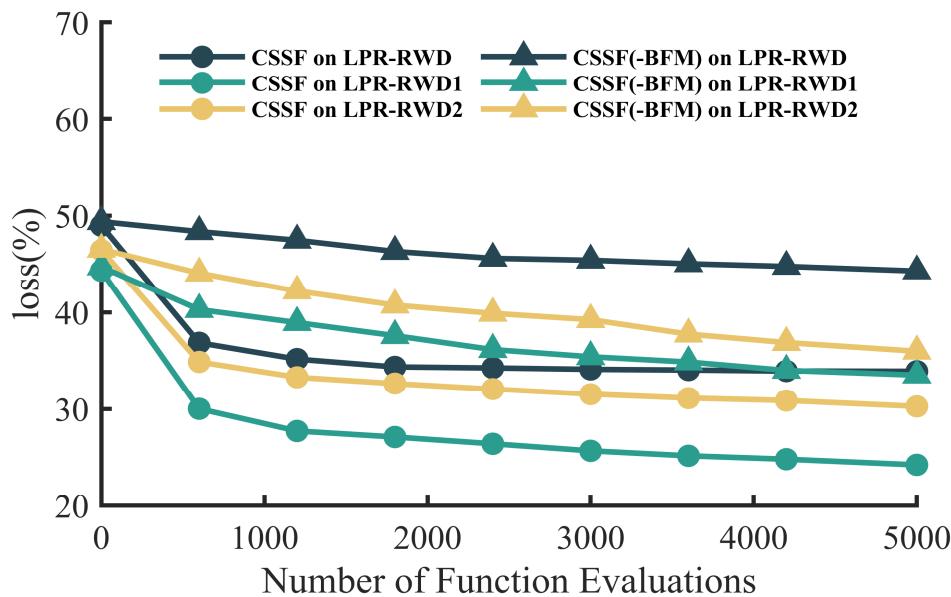
Table 6 presents the results of the Friedman test rankings, highlighting that the CSSF algorithm proposed in this study consistently achieves the top rank across both real-world and synthetic datasets. Its performance remains consistently effective across different datasets, demonstrating superiority over other comparative algorithms. Specifically, the EESHHO and GEO algorithms exhibit better performance compared to the MSEA and MFOSPEA2 algorithms. Importantly, the results from synthetic datasets closely align with those from real-world datasets, as indicated by a low standard deviation of the rank values between these two types of datasets. This suggests that synthetic datasets can effectively substitute real-world datasets in initial experiments assessing algorithm performance, given their similar data characteristics.

In summary, the CSSF algorithm proposed in this study showcases robust capabilities in real learning path recognition, thereby confirming its superior effectiveness in algorithmic performance.

### 5.5. Ablation Experiment

In this phase of the experiment, the CSSF algorithm, designed based on a positive and negative feedback strategy, will be compared with the CSSF(-BFM) algorithm, which excludes this strategy. The objective is to validate the effectiveness of the optimization strategy introduced in this paper on the CSSF(-BFM) algorithm. The experiments utilize real-world datasets from the LPR-RWD series to provide a comprehensive assessment of algorithmic performance.

The CSSF algorithm proposed in this study incorporates a positive and negative feedback mechanism to enhance the efficiency of learning path recognition. To elucidate the reasons behind its superior performance, this experiment contrasts the CSSF algorithm with and without the bidirectional feedback mechanism (CSSF(-BFM)). Figure 4 presents an analysis of the change in loss values after 5000 runs for both algorithms across the LPR-RWD series of problems. To ensure the robustness of the experimental results, each algorithm underwent multiple repetitions across these three test problems, with the average taken as the final experimental result to mitigate the impact of parameter variations on the algorithm's path recognition capabilities within different datasets. This approach helps ascertain whether the bidirectional feedback mechanism directly contributes to the algorithm's performance enhancement. The experimental findings indicate that the initial number of knowledge components significantly influences the initial population's loss values. Specifically, fewer knowledge components lead to lower initial loss values in the population, which closely correlates with the algorithm's workload. As the number of knowledge components increases, the initial population generated by the algorithm may exhibit more errors, thereby increasing the average initial loss value of the population. This observation suggests that an increase in the number of knowledge components adversely affects the algorithm's initial performance, necessitating optimization strategies to mitigate these effects. Detailed analysis of the experimental data reveals that the CSSF algorithm substantially improves its recognition efficiency and path optimization capability through the incorporation of positive and negative feedback mechanisms. This bidirectional feedback mechanism enables the algorithm to dynamically adjust and optimize learning paths more effectively. In the early stages with fewer iterations, the CSSF algorithm incorporating the feedback mechanism demonstrates rapid decreases in loss values, while over subsequent iterations, it continues to achieve steady reductions in loss values. In contrast, the CSSF(-BFM) algorithm, lacking the bidirectional feedback mechanism, exhibits inferior efficiency in recognition and optimization, characterized by slower rates of reduction in loss values and less stable final values.



**Figure 4.** The algorithm loss distribution plot. The variation of loss values of the CSSF algorithm that preserves the bidirectional feedback mechanism and removes the bidirectional feedback mechanism is presented in this plot.

In conclusion, this experimental phase underscores the pivotal role of the positive and negative feedback mechanisms in enhancing the CSSF algorithm's performance. By comparing it with the CSSF(-BFM) algorithm, the results unequivocally highlight the direct contribution of the bidirectional feedback mechanism to algorithmic effectiveness. This finding provides compelling evidence for further refining and implementing the CSSF algorithm, guiding future research and development of related algorithms. Applied to real-world scenarios, the CSSF algorithm demonstrates robust capabilities in learning path recognition and offers effective solutions for managing complex datasets.

In this experimental phase, we conducted a comparison between the CSSF algorithm, which integrates a positive and negative feedback strategy, and the CSSF(-BFM) algorithm without this strategy. The objective was to validate the effectiveness of the introduced optimization strategy using the LPR-RWD series of problems derived from real-world datasets, ensuring a comprehensive evaluation of algorithmic performance. Figure 4 depicts the performance of the CSSF and CSSF(-BFM) algorithms with circles and triangles, respectively. Initially, there is minimal disparity in the fitness levels of their initial populations. However, as the number of iterations progresses, notable differences emerge: the CSSF(-BFM) algorithm exhibits slower improvement compared to the CSSF algorithm.

Despite similar initial loss values, the CSSF algorithm demonstrates robust optimization capabilities throughout the iterative process, achieving faster convergence. Figure 4 utilizes deep blue, green, and yellow colors to illustrate the optimization trajectories of both algorithms across three scenarios: LPR-RWD, LPR-RWD1, and LPR-RWD2, featuring knowledge point dimensions of 116, 50, and 50, respectively. Across all these scenarios, the CSSF algorithm, equipped with the bidirectional feedback mechanism, consistently demonstrates superior performance during iterative optimization. Specifically, in LPR-RWD, after 5000 runs, the final loss value of the CSSF algorithm is approximately 23% lower than that of the CSSF(-BFM) algorithm. In LPR-RWD1, this difference increases to approximately 28%, and in LPR-RWD2, it stands at approximately 16%. These findings underscore the significant positive impact of the bidirectional feedback mechanism on learning path determination in real educational datasets.

In conclusion, this phase of the experiment validates the pivotal role of the positive and negative feedback strategy in enhancing the CSSF algorithm's performance. The results highlight its effectiveness in optimizing learning paths across diverse dataset dimensions,

providing valuable insights for further refinement and application of the CSSF algorithm in educational contexts.

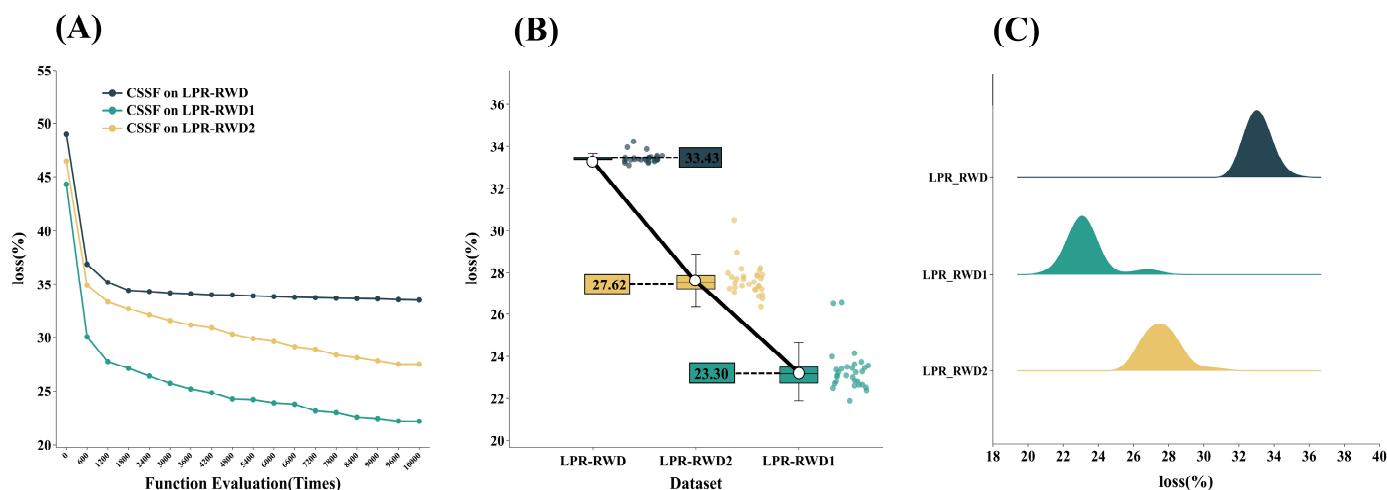
In general, due to the intricate nature of optimization in Learning Path Recognition (LPR) problems, the CSSF(-BFM) algorithm exhibits comparatively lower effectiveness than the CSSF algorithm. Conversely, the proposed CSSF algorithm demonstrates substantial advantages in tackling these challenges, primarily attributed to the incorporation of a positive and negative feedback mechanism. This mechanism plays a pivotal role in optimizing the learning path recognition process, facilitating rapid reduction of loss values over multiple iterations and achieving faster convergence compared to algorithms lacking such a mechanism.

Upon analyzing the experimental data, it becomes evident that the integration of the positive and negative feedback mechanisms significantly enhances the performance of the CSSF algorithm. This enhancement translates into enhanced optimization capabilities and reduced loss values across diverse scenarios. These findings not only validate the efficacy of the optimization strategy proposed in this study but also establish a robust theoretical basis for further advancements and applications of the CSSF algorithm in future research. The CSSF algorithm demonstrates robust efficiency in learning path recognition and optimization, particularly when dealing with complex datasets, thereby establishing itself as a potent tool for addressing real-world challenges.

### 5.6. Convergence Experiment

To analyze the convergence characteristics of the proposed algorithm, detailed experiments were conducted in this study. Specifically focusing on stochastic search algorithms, such as evolutionary algorithms, which often exhibit instability and risk of identification errors in multi-dimensional optimization problems due to their stochastic nature and inherent randomness. To rigorously evaluate the stability and performance of the CSSF algorithm in the context of Learning Path Recognition (LPR) problems, we conducted experimental analyses using real datasets in the LPR-RWD series. This section provides a comprehensive description of the convergence experiment results of the CSSF algorithm in the LPR-RWD series problems. The parameter settings used in these experiments were consistent with those outlined in Table 3. The analysis covers the convergence of the CSSF algorithm across three distinct problems: LPR-RWD1, LPR-RWD2, and LPR-RWD3. Figure 5A illustrates the convergence curves of the CSSF algorithm on these problems, with deep blue, green, and orange colors representing the distribution of loss values for LPR-RWD1, LPR-RWD2, and LPR-RWD3, respectively. Each problem underwent 10,000 population iterations to assess algorithmic convergence. From Figure 5A, it is evident that the CSSF algorithm performs most effectively on LPR-RWD1, followed by LPR-RWD2, while its performance on LPR-RWD3 is comparatively lower. This disparity can be attributed to the heightened complexity of LPR-RWD3, which necessitates the identification of a larger number of knowledge components compared to the other two problems. To further scrutinize the algorithm's performance, box plots were generated to display the results of 30 runs of the CSSF algorithm on each of the LPR-RWD series problems, as depicted in Figure 5B. The median values in the box plots delineate the distribution of results, allowing for a comparative assessment of stability across different problems. Additional comparative analysis from the experimental data in Figure 5B reveals that the average loss values of the CSSF algorithm on the LPR-RWD series problems are 33.43%, 24.62%, and 23.30%, respectively. These results closely align with those presented in Table 3, with differences in average loss values not exceeding  $\pm 2\%$  between the two sets of experiments. This consistency provides robust validation of the CSSF algorithm's convergence. Figure 5B also includes scatter plot distributions of CSSF algorithm results on LPR-RWD problems, illustrating the algorithm's convergence and consistency visually. The x-axis depicts the algorithm's loss values across different problems, highlighting that the CSSF algorithm achieves superior convergence on LPR-RWD1, followed by LPR-RWD2 and LPR-RWD3. Particularly on LPR-RWD2 and LPR-

RWD3, the distribution of loss values is more concentrated, underscoring the algorithm's strong performance consistency.



**Figure 5.** Convergence analysis. (A) shows the evolution curves of CSSF on the real-world LPR-RWD series problems. (B) presents the convergence analysis box plots of the CSSF algorithm on real-world datasets. (C) illustrates the degrees of aggregation of the CSSF algorithm on real-world datasets using ridge plots. The purpose of (A–C) is to analyze the convergence and aggregation of the CSSF algorithm on different problems.

In conclusion, these experiments demonstrate that the CSSF algorithm exhibits outstanding convergence and stability in Learning Path Recognition problems. The results of the LPR-RWD series problems underscore the algorithm's capability to maintain stability and consistency across complex scenarios. Whether in single or repeated experiments, the CSSF algorithm consistently performs well, reaffirming its efficacy as a robust evolutionary algorithm for addressing multi-dimensional optimization challenges.

## 6. Discussion

In this study, we propose a novel graph structure search algorithm based on a positive and negative feedback mechanism for analyzing learning paths. Through a series of experiments, we evaluated the performance of this algorithm and compared it with four existing algorithms: MSEA, GEO, EESHHO, and MFOSPEA2. We conducted comparative experiments by applying the CSSF algorithm alongside MSEA, GEO, EESHHO, and MFOSPEA2 on both synthetic and real datasets. The results demonstrate that the CSSF algorithm excels in handling large-scale learning path graphs in both synthetic and real datasets. Specifically, the CSSF algorithm shows superior performance and robustness compared to the benchmark algorithms, especially when the number of nodes and edges is substantial.

The primary contributions of this work are outlined as follows:

1. We introduce a multi-subgroup collaborative search mechanism aimed at enhancing search efficiency. By categorizing individuals within the population into superior, exploratory, and elimination subgroups based on their fitness values, these subgroups are tasked with maintaining individual-level superiority, fostering population-level diversity, and ensuring simplicity in the solution set. This approach significantly improves search efficiency.
2. A designed bidirectional feedback mechanism is implemented to distinguish high-quality from low-quality edges within the graph, thereby guiding the graph structure search process. This mechanism addresses optimization challenges arising from the sparse nature of edges in the Learning Path Recognition (LPR) task by oversampling

- identified high-quality edges and undersampling low-quality edges at the population level.
3. Based on the aforementioned work, the CSSF framework is proposed to facilitate the sustainable learning and evolution of the proposed algorithm within continuously updated educational data. Through experimental validation on real-world datasets, we demonstrate the efficacy of CSSF. The project's source code is openly available on GitHub at <https://github.com/YuanHao-CS/CSSF> for further exploration by interested readers. This link was accessed on 19 June 2024.

In terms of learning path identification, our algorithm effectively finds the optimal learning path with high efficiency. This advantage is mainly attributed to the positive and negative feedback adjustment mechanism, which allows the algorithm to dynamically adjust the generation of learning paths, thereby reducing unnecessary computations. Our algorithm is particularly suitable for analyzing learning path graphs with a large number of nodes and complex relationships, such as course recommendations on online learning platforms and educational data mining. In these applications, our algorithm can efficiently identify the optimal learning paths and provide personalized learning suggestions.

## 7. Recommendations

As a foundational study advancing adaptive learning systems, this research establishes a technical framework for uncovering relationships within knowledge structures. Building on this study, the proposed algorithmic framework can be further enhanced and optimized to better support the sustainability of AI in education. First, beyond effectively identifying learning paths, there should be an emphasis on the explainability of these paths. The education sector highly values the explainability of technology, not just its efficacy. Therefore, it is crucial to enhance the explainability of knowledge learning paths [63]. Second, after identifying students' weak knowledge structures, large language models (LLMs) can be utilized to automatically generate learning materials to assist students in overcoming their difficulties. The rapid development of LLMs has significantly transformed education, with LLMs already capable of effectively generating learning materials for specific subjects, such as mathematics [64]. Third, by analyzing the relationships between students' knowledge components and integrating LLMs, guided teaching can be further realized to improve learning outcomes. Guided learning based on LLMs or LLM-supported multi-agent systems can effectively help learners better master subject knowledge and enhance adaptive learning systems [65]. Based on LLMs, students can also receive assistance with career planning [66]. Fourth, LLMs can also be employed to achieve learning data annotation and feedback report generation, thereby lowering the usage threshold of this framework and making the proposed technology more accessible to teachers. Students' knowledge structure data are complex and challenging for humans to interpret directly. Machine learning and data mining techniques can aid in predicting or analyzing students' performance and learning data, thereby assisting educators and educational decision-makers in understanding these complexities [67]. By incorporating LLMs for learning data annotation and feedback report generation, teachers can quickly understand students' learning statuses and develop targeted instructional activities [68]. We deeply appreciate the rapid advancements in AI for education and eagerly anticipate future developments in educational technology.

## 8. Limitations

The CSSF algorithm efficiently identifies optimal learning paths and provides personalized learning recommendations. However, the algorithm's performance may be affected when dealing with highly dynamic learning path graphs. Additionally, further optimization is needed to improve computational efficiency when handling extremely large datasets, such as those with millions of nodes and edges. Integrating the CSSF algorithm into adaptive learning systems also requires that teachers undergo training to use the proposed framework effectively. Therefore, we plan to develop the algorithm and the

adaptive learning system for broader usability in the future. This includes enhancing the study with an easy-to-use GUI interface to make it more user-friendly. We also aim to verify the effectiveness and applicability of the algorithm in different educational scenarios. Further research can explore the application of the algorithm in other fields, such as social network analysis and recommendation systems, to expand its range of applications.

## 9. Conclusions

This paper proposes a cooperative co-evolutionary algorithm based on a multi-subgroup collaboration strategy, referred to as CSSF. By subjecting different individuals to either positive or negative environmental changes based on their fitness values, this algorithm effectively alters their states, thereby significantly improving the efficiency of learning path identification. The CSSF framework demonstrates its unique contributions and practical value in several aspects.

The CSSF framework enhances the algorithm's adaptability to diverse datasets through its multi-subgroup collaboration strategy. This approach not only improves the accuracy of learning path identification but also significantly maintains population diversity, preventing the common pitfall of traditional algorithms falling into local optima. Experimental validation shows that the CSSF algorithm achieves a noticeable improvement in identification efficiency on generated datasets compared to existing algorithms, indicating that CSSF can identify learning paths more quickly and accurately. In practical applications, the advantages of the CSSF algorithm are particularly evident. Experimental results indicate that CSSF also exhibits superior performance on real datasets, demonstrating its robustness and efficiency in different data environments. This implies that the CSSF algorithm is not only suitable for laboratory testing conditions but can also play a crucial role in actual educational environments. By identifying and understanding students' learning paths, CSSF can be integrated with intelligent education systems to provide personalized educational programs, better adapting to and meeting individual student needs, thereby improving educational quality. The contributions of the CSSF algorithm to the sustainability of education are noteworthy. Utilizing big data and machine learning technologies, CSSF can accurately identify key factors affecting learning outcomes, optimizing course design and teaching methods. The application of intelligent feedback systems and real-time evaluation models allows the educational process to quickly respond to student needs, providing timely suggestions and support, ensuring that every student can learn in their optimal state. This not only enhances the overall effectiveness of education but also lays a solid foundation for the sustainable development of education. By applying learning path identification algorithms to adaptive learning systems, it is possible to dynamically adjust teaching content and learning paths based on each student's progress and understanding. This personalized learning experience helps struggling students learn at a pace suited to them, while also providing more challenging content for advanced students. In this way, every student can achieve maximum development at their own learning level, reducing the inequalities brought about by a uniform pace of teaching.

In summary, the CSSF framework, through its innovative application of a multi-subgroup collaboration strategy in learning path identification, not only improves identification efficiency and accuracy but also demonstrates its robustness and efficiency in different data environments. In practical educational applications, CSSF provides strong support for intelligent education systems, promoting the development of personalized education. Additionally, CSSF's contributions to optimizing course design and teaching methods offer new insights and approaches for the sustainable development of education.

**Author Contributions:** Conceptualization, T.-Y.L., Y.-H.J. and Y.W.; Data curation, T.-Y.L.; Methodology, T.-Y.L. and Y.-H.J.; Project administration, Y.-H.J. and X.W.; Software, T.-Y.L.; Supervision, X.W. and S.H.; Validation, Y.W.; Visualization, T.-Y.L.; Writing—original draft, T.-Y.L., Y.-H.J. and Y.W.; Writing—review and editing, X.W. and L.D. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported in part by the Postgraduate Research & Practice Innovation Program of Jiangsu Province (SJCX24\_2545), and the Special Foundation for Interdisciplinary Talent Training in “AI Empowered Psychology/Education”, under the Grant 2024JRC-03, with the project titled “Multi-Agent Driven Mathematical Knowledge Causal Diagnosis and Learning Platform”, and the Doctoral Research and Innovation Foundation of the School of Computer Science and Technology, East China Normal University, under the title of “Intrinsic Mechanisms of Collaborative Learning Achievements Formation and Their Interpretability Analysis: from Game Theory Simulation, AI Agent Simulation to Empirical Analysis”. Meanwhile, Jicong Duan is acknowledged by the authors for his contributions to data processing and visualization in this research, affiliated with the School of Computer and the School of Automation at Jiangsu University of Science and Technology.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data are openly available on CodaLab at <https://eedi.com/projects/neurips-2022>, as referenced in [50] of our manuscript, which was accessed on 20 July 2022. Additionally, the code used in this study is available on Github at <https://github.com/YuanHao-CS/CSSF>, as referenced in [62] of our manuscript, which was accessed on 19 June 2024.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. UNESCO. *Reimagining Our Futures Together: A New Social Contract for Education*; Educational and Cultural Organization of the United Nations: Paris, France, 2021.
2. Alrakhawi, H.A.; Jamiat, N.; Abu-Naser, S.S. Intelligent Tutoring Systems in Education: A Systematic Review of Usage, Tools, Effects and Evaluation. *J. Theor. Appl. Inf. Technol.* **2023**, *101*, 1205–1226.
3. Gligoreia, I.; Cioca, M.; Oancea, R.; Gorski, A.-T.; Gorski, H.; Tudorache, P. Adaptive Learning Using Artificial Intelligence in E-Learning: A Literature Review. *Educ. Sci.* **2023**, *13*, 1216. [[CrossRef](#)]
4. Shu, X.; Ye, Y. Knowledge Discovery: Methods from Data Mining and Machine Learning. *Soc. Sci. Res.* **2023**, *110*, 102817. [[CrossRef](#)] [[PubMed](#)]
5. Zhou, Z.; Liu, G.; Tang, Y. Multi-Agent Reinforcement Learning: Methods, Applications, Visionary Prospects, and Challenges. *arXiv* **2023**, arXiv:2305.10091.
6. Rančić, D.; Kuk, K.; Pronić-Rančić, O.; Randelović, D. Agent-Based Approach for Game-Based Learning Applications: Case Study in Agent-Personalized Trend in Engineering Education. In *Agent and Multi-Agent Systems: Technologies and Applications*; Jezic, G., Howlett, R.J., Jain, L.C., Eds.; Smart Innovation, Systems and Technologies; Springer International Publishing: Cham, Switzerland, 2015; Volume 38, pp. 453–466, ISBN 978-3-319-19727-2.
7. Vallée, A.; Blacher, J.; Cariou, A.; Sorbets, E. Blended Learning Compared to Traditional Learning in Medical Education: Systematic Review and Meta-Analysis. *J. Med. Internet Res.* **2020**, *22*, e16504. [[CrossRef](#)]
8. Beyene, W.M.; Mekonnen, A.T.; Giannoumis, G.A. Inclusion, Access, and Accessibility of Educational Resources in Higher Education Institutions: Exploring the Ethiopian Context. *Int. J. Incl. Educ.* **2023**, *27*, 18–34. [[CrossRef](#)]
9. Pearl, J. *Causality*; Cambridge University Press: Cambridge, UK, 2009.
10. Sun, K.; Liu, Y.; Guo, Z.; Wang, C. Visualization for Knowledge Graph Based on Education Data. *Int. J. Softw. Inform.* **2016**, *10*, 1–13.
11. Kosari, S.; Rao, Y.; Jiang, H.; Liu, X.; Wu, P.; Shao, Z. Vague Graph Structure with Application in Medical Diagnosis. *Symmetry* **2020**, *12*, 1582. [[CrossRef](#)]
12. Parisi, F.; Ruggieri, S.; Lovreglio, R.; Fanti, M.P.; Uva, G. On the Use of Mechanics-Informed Models to Structural Engineering Systems: Application of Graph Neural Networks for Structural Analysis. In *Proceedings of the Structures*; Elsevier: Amsterdam, The Netherlands, 2024; Volume 59, p. 105712.
13. Tang, J.; Yang, Y.; Wei, W.; Shi, L.; Su, L.; Cheng, S.; Yin, D.; Huang, C. GraphGPT: Graph Instruction Tuning for Large Language Models. *arXiv* **2023**, arXiv:2310.13023.
14. Ding, X.; Xia, C.; Zhang, X.; Chu, X.; Han, J.; Ding, G. Repmlp: Re-Parameterizing Convolutions into Fully-Connected Layers for Image Recognition. *arXiv* **2021**, arXiv:2105.01883.
15. Zhuang, F.; Qi, Z.; Duan, K.; Xi, D.; Zhu, Y.; Zhu, H.; Xiong, H.; He, Q. A Comprehensive Survey on Transfer Learning. *Proc. IEEE* **2019**, *109*, 43–76. [[CrossRef](#)]
16. Liu, Z.; Wang, Y.; Vaidya, S.; Ruehle, F.; Halverson, J.; Soljačić, M.; Hou, T.Y.; Tegmark, M. Kan: Kolmogorov-Arnold Networks. *arXiv* **2024**, arXiv:2404.19756.
17. Chickering, D.M. Optimal Structure Identification with Greedy Search. *J. Mach. Learn. Res.* **2002**, *3*, 507–554.
18. Cormen, T.H.; Leiserson, C.E.; Rivest, R.L.; Stein, C. *Introduction to Algorithms*; MIT Press: Cambridge, MA, USA, 2022.
19. Tsamardinos, I.; Brown, L.E.; Aliferis, C.F. The Max-Min Hill-Climbing Bayesian Network Structure Learning Algorithm. *Mach. Learn.* **2006**, *65*, 31–78. [[CrossRef](#)]

20. Tian, Y.; He, C.; Cheng, R.; Zhang, X. A Multistage Evolutionary Algorithm for Better Diversity Preservation in Multiobjective Optimization. *IEEE Trans. Syst. Man Cybern Syst.* **2021**, *51*, 5880–5894. [[CrossRef](#)]
21. Mohammadi-Balani, A.; Dehghan Nayeri, M.; Azar, A.; Taghizadeh-Yazdi, M. Golden Eagle Optimizer: A Nature-Inspired Metaheuristic Algorithm. *Comput. Ind. Eng.* **2021**, *152*, 107050. [[CrossRef](#)]
22. Lewis, H.S. *Leaders and Followers: Some Anthropological Perspectives*; Addison-Wesley: Reading, MA, USA, 1974.
23. Rubin, D.B. Estimating Causal Effects of Treatments in Randomized and Nonrandomized Studies. *J. Educ. Psychol.* **1974**, *66*, 688. [[CrossRef](#)]
24. Wang, D.; Chen, D. Causal Inference: Origin and Development. *Control. Eng. China* **2022**, *29*, 464–473.
25. Pearl, J.; Mackenzie, D. AI Can't Reason Why. *Wall Street J.* **2018**. Available online: <https://www.wsj.com/articles/ai-cant-reason-why-1526657442> (accessed on 1 July 2024).
26. Pearl, J. Causal Inference. *Causality Object. Assess.* **2010**, *6*, 39–58.
27. Spirtes, P.; Zhang, K. Causal Discovery and Inference: Concepts and Recent Methodological Advances. In *Proceedings of the Applied Informatics*; Springer: Berlin/Heidelberg, Germany, 2016; Volume 3, pp. 1–28.
28. Le, T.D.; Hoang, T.; Li, J.; Liu, L.; Liu, H.; Hu, S. A Fast PC Algorithm for High Dimensional Causal Discovery with Multi-Core PCs. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **2016**, *16*, 1483–1495. [[CrossRef](#)]
29. Entner, D.; Hoyer, P.O. On Causal Discovery from Time Series Data Using FCI. *Probabilistic Graph. Models* **2010**, *16*, 121–128.
30. Ramsey, J.; Glymour, M.; Sanchez-Romero, R.; Glymour, C. A Million Variables and More: The Fast Greedy Equivalence Search Algorithm for Learning High-Dimensional Graphical Causal Models, with an Application to Functional Magnetic Resonance Images. *Int. J. Data Sci. Anal.* **2017**, *3*, 121–129. [[CrossRef](#)] [[PubMed](#)]
31. Suzuki, J. A Theoretical Analysis of the BDeu Scores in Bayesian Network Structure Learning. *Behaviormetrika* **2017**, *44*, 97–116. [[CrossRef](#)]
32. Kuha, J. AIC and BIC: Comparisons of Assumptions and Performance. *Sociol. Methods Res.* **2004**, *33*, 188–229. [[CrossRef](#)]
33. Siripatana, A.; Mayo, T.; Srivastava, I.; Knio, O.; Dawson, C.; Le Maitre, O.; Hoteit, I. Assessing an Ensemble Kalman Filter Inference of Manning's n Coefficient of an Idealized Tidal Inlet against a Polynomial Chaos-Based MCMC. *Ocean Dyn.* **2017**, *67*, 1067–1094. [[CrossRef](#)]
34. Jiang, B.; Wei, Y.; Gu, M.; Yin, C. Understanding Students' Backtracking Behaviors in Digital Textbooks: A Data-Driven Perspective. *Interact. Learn. Environ.* **2023**, *1*–18. [[CrossRef](#)]
35. Jiang, B.; Wei, Y.; Zhang, T.; Zhang, W. Improving the Performance and Explainability of Knowledge Tracing via Markov Blanket. *Inf. Process. Manag.* **2024**, *61*, 103620. [[CrossRef](#)]
36. Spirtes, P.; Richardson, T.; Meek, C. Heuristic Greedy Search Algorithms for Latent Variable Models. In *Proceedings of the AI & STAT'97*, Fort Lauderdale, FL, USA, 24 November 1997; pp. 481–488.
37. Kitano, H. Designing Neural Networks Using Genetic Algorithms with Graph Generation System. *Complex Syst.* **1990**, *4*, 461–476.
38. Fahlman, S.; Lebiere, C. The Cascade-Correlation Learning Architecture. *Adv. Neural Inf. Process. Syst.* **1989**, *2*, 524–532.
39. Real, E.; Aggarwal, A.; Huang, Y.; Le, Q.V. Regularized Evolution for Image Classifier Architecture Search. *Proc. AAAI Conf. Artif. Intell.* **2019**, *33*, 4780–4789. [[CrossRef](#)]
40. Wei, X.-F.; Tang, K.; Chen, Z.-W.; Chen, H.-J.; Shi, Y.-H.; Jiang, Y.-H. ACDO: An Ant Colony Dynamic Optimization Framework for Tourism Route Planning. In *Proceedings of the 2023 4th International Conference on Computer Science and Management Technology*, Xi'an, China, 13–15 October 2023; Association for Computing Machinery: New York, NY, USA, 2024; pp. 851–856.
41. Romera-Paredes, B.; Barekatain, M.; Novikov, A.; Balog, M.; Kumar, M.P.; Dupont, E.; Ruiz, F.J.R.; Ellenberg, J.S.; Wang, P.; Fawzi, O.; et al. Mathematical Discoveries from Program Search with Large Language Models. *Nature* **2023**, *625*, 468–475. [[CrossRef](#)] [[PubMed](#)]
42. Zhang, Y.; Zheng, Q.-Q.; He, L.-J.; Tian, H.-W. Ship Traffic Optimization Method for Solving the Approach Channel and Lock Co-Scheduling Problem of the Three Gorges Dam on the Yangzi River. *Ocean Eng.* **2023**, *276*, 114196. [[CrossRef](#)]
43. Tang, K.; Wei, X.-F.; Jiang, Y.-H.; Chen, Z.-W.; Yang, L. An Adaptive Ant Colony Optimization for Solving Large-Scale Traveling Salesman Problem. *Mathematics* **2023**, *11*, 4439. [[CrossRef](#)]
44. Stanley, K.O.; Miikkulainen, R. Evolving Neural Networks through Augmenting Topologies. *Evol. Comput.* **2002**, *10*, 99–127. [[CrossRef](#)] [[PubMed](#)]
45. Gomez, F.J.; Miikkulainen, R. Solving Non-Markovian Control Tasks with Neuroevolution. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI99)*, Stockholm, Sweden, 31 July–6 August 1999; Volume 99, pp. 1356–1361.
46. Salimans, T.; Ho, J.; Chen, X.; Sidor, S.; Sutskever, I. Evolution Strategies as a Scalable Alternative to Reinforcement Learning. *arXiv* **2017**, arXiv:1703.03864.
47. Zoph, B.; Vasudevan, V.; Shlens, J.; Le, Q.V. Learning Transferable Architectures for Scalable Image Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8697–8710.
48. Zoph, B.; Le, Q.V. Neural Architecture Search with Reinforcement Learning. *arXiv* **2016**, arXiv:1611.01578.
49. Real, E.; Moore, S.; Selle, A.; Saxena, S.; Suematsu, Y.L.; Tan, J.; Le, Q.V.; Kurakin, A. Large-Scale Evolution of Image Classifiers. In *Proceedings of the International Conference on Machine Learning*; PMLR: Birmingham, UK, 2017; pp. 2902–2911.
50. Gong, W.; Smith, D.; Wang, Z.; Barton, C.; Pawłowski, N.; Jennings, J.; Zhang, C. Instructions and Guide: Causal Insights for Learning Paths in Education. CodaLab. 2024. Available online: <https://eedi.com/projects/neurips-2022> (accessed on 1 July 2024).

51. Blot, A.; Petke, J. Empirical Comparison of Search Heuristics for Genetic Improvement of Software. *IEEE Trans. Evol. Comput.* **2021**, *25*, 1001–1011. [[CrossRef](#)]
52. Chen, S.; Wang, W.; Xia, B.; You, X.; Peng, Q.; Cao, Z.; Ding, W. CDE-GAN: Cooperative Dual Evolution-Based Generative Adversarial Network. *IEEE Trans. Evol. Comput.* **2021**, *25*, 986–1000. [[CrossRef](#)]
53. Jiang, Y.-H.; Gao, S.; Yin, Y.-H.; Xu, Z.-F.; Wang, S.-Y. A Control System of Rail-Guided Vehicle Assisted by Transdifferentiation Strategy of Lower Organisms. *Eng. Appl. Artif. Intell.* **2023**, *123*, 106353. [[CrossRef](#)]
54. Siddique, A.; Browne, W.N.; Grimshaw, G.M. Frames-of-Reference Based Learning: Overcoming Perceptual Aliasing in Multi-Step Decision Making Tasks. *IEEE Trans. Evol. Comput.* **2021**, *26*, 174–187. [[CrossRef](#)]
55. Tang, K.; Liu, S.; Yang, P.; Yao, X. Few-Shots Parallel Algorithm Portfolio Construction via Co-Evolution. *IEEE Trans. Evol. Comput.* **2021**, *25*, 595–607. [[CrossRef](#)]
56. Heckerman, D.; Meek, C.; Cooper, G. A Bayesian Approach to Causal Discovery. *Innov. Mach. Learn. Theory Appl.* **2006**, *138*, 1–28.
57. Gao, Z.; Ren, B. Network Learning Based on Ant Colony Optimization. *Syst. Eng. Electron.* **2010**, *6*, 4.
58. Yang, C.; Ji, J.; Liu, J.; Liu, J.; Yin, B. Structural Learning of Bayesian Networks by Bacterial Foraging Optimization. *Int. J. Approx. Reason.* **2016**, *69*, 147–167. [[CrossRef](#)]
59. Tian, Y.; Zhu, W.; Zhang, X.; Jin, Y. A Practical Tutorial on Solving Optimization Problems via PlatEMO. *Neurocomputing* **2023**, *518*, 190–205. [[CrossRef](#)]
60. Jiao, R.; Xue, B.; Zhang, M. A Multiform Optimization Framework for Constrained Multiobjective Optimization. *IEEE Trans. Cybern.* **2023**, *53*, 5165–5177. [[CrossRef](#)] [[PubMed](#)]
61. Jangir, P.; Heidari, A.A.; Chen, H. Elitist Non-Dominated Sorting Harris Hawks Optimization: Framework and Developments for Multi-Objective Problems. *Expert Syst. Appl.* **2021**, *186*, 115747. [[CrossRef](#)]
62. Liu, T.-Y.; Jiang, Y.-H.; Wei, Y.; Wang, X.; Huang, S.; Dai, L. CSSF: A General-Purpose Algorithm Framework Designed to Recognition Real-World Educational Learning Path. Github. 2024. Available online: <https://github.com/YuanHao-CS/CSSF> (accessed on 1 July 2024).
63. Wei, Y.; Zhou, Y.; Jiang, Y.-H.; Jiang, B. Enhancing Explainability of Knowledge Learning Paths: Causal Knowledge Networks. *arXiv* **2024**, arXiv:2406.17518. [[CrossRef](#)]
64. Li, R.; Wang, Y.; Zheng, C.; Jiang, Y.-H.; Jiang, B. Generating Contextualized Mathematics Multiple-Choice Questions Utilizing Large Language Models. In *Artificial Intelligence in Education. Posters and Late Breaking Results, Workshops and Tutorials, Industry and Innovation Tracks, Practitioners, Doctoral Consortium and Blue Sky*; Olney, A.M., Chounta, I.-A., Liu, Z., Santos, O.C., Bittencourt, I.I., Eds.; Springer Nature: Cham, Switzerland, 2024; pp. 494–501.
65. Jiang, Y.-H.; Li, R.; Zhou, Y.; Qi, C.; Hu, H.; Wei, Y.; Jiang, B.; Wu, Y. AI Agent for Education: Von Neumann Multi-Agent System Framework. In Proceedings of the 28th Global Chinese Conference on Computers in Education (GCCCE 2024); Global Chinese Conference on Computers in Education, Chongqing, China, 1–5 June 2024; pp. 77–84. Available online: [http://gccce2024.swu.edu.cn/GCCCE2024\\_gongzuofanglunwenji2024-06-23A.pdf#page=95](http://gccce2024.swu.edu.cn/GCCCE2024_gongzuofanglunwenji2024-06-23A.pdf#page=95) (accessed on 4 July 2024).
66. Zhou, Y.; Zhang, Y.; Liu, N. Research on the Design of an AI Career Path Recommendation System Based on MBTI from a Cross-Cultural Perspective. *Artif. Intell. Technol. Res.* **2024**, *2*, 1.
67. Zhou, Y.; Li, Z. Utilizing Machine Learning Algorithms for Predictive Analysis of Student Performance: A Database-Integrated Approach. *Int. J. Math. Syst. Sci.* **2024**, *6*, 6.
68. Zhou, Y.; Zhang, M.; Jiang, Y.-H.; Liu, N.; Jiang, B. A Study on Educational Data Analysis and Personalized Feedback Report Generation Based on Tags and ChatGPT. In Proceedings of the 28th Global Chinese Conference on Computers in Education (GCCCE 2024); Global Chinese Conference on Computers in Education, Chongqing, China, 1–5 June 2024; pp. 108–115. Available online: [http://gccce2024.swu.edu.cn/GCCCE2024\\_gongzuofanglunwenji2024-06-23A.pdf#page=126](http://gccce2024.swu.edu.cn/GCCCE2024_gongzuofanglunwenji2024-06-23A.pdf#page=126) (accessed on 4 July 2024).

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.