JOURNAL LA MULTIAPP

VOL. 06, ISSUE 03 (457-469), 2025
DOI: 10.37899/journallamultiapp.v6i3.1991

Optimizing Learning Object Sequencing in E-Learning Systems Using Human Behaviour-Based Particle Swarm Optimization

Raed Waheed Kadhim[1]

[1]*Department of Computer Science, College of Science, Mustansiriyah University, Baghdad, Iraq*

**Corresponding Author: Raed Waheed Kadhim*
Email: raedwk.cs@uomustansiriyah.edu.iq

Article Info

Article history:
Received 31 January 2025
Received in revised form 19 February 2025
Accepted 30 April 2025

Keywords:
E-Learning Optimization
LO
Competency-Based Sequencing
Constraint Satisfaction Problem
HPSO

Abstract

The distinctive requirements, educational attainment, and learning response of the learners are the critical key issues in the e-learning system. This goal is achieved by identifying the students' diverse assessments of their identity and capabilities and assigning them appropriate learning materials, as indicated by these highlights. The present paper introduces an efficient learning system that optimizes the sequencing of Learning Objects (LOs) in an e-learning system. Learning Objects (LOs) are educational materials typically divided into components. The procedure is performed by sequencing the learning objects of pupils or learners, and the sequence represents the organized arrangement of LOs. The sequencing problem can be considered a Constraint Satisfaction Problem (CSP) due to the utilization of the competency to characterize the correlation among LOs. The Human Behavior-based Particle Swarm Optimization (HPSO) algorithm can be employed to solve the problem using a swarm intelligence scheme. Results indicate that the algorithm is more effective in resolving the matter.

Introduction

Within an e-learning system, Learning Objects (LOs) illustrate all actions (Alnawas et al., 2022). The method encourages the creation of learning objects (LOs) as minimal educational units. Subsequently, the learning objectives are gathered and further consolidated to create larger instructional units (Govindasamy, 2001; Malalla & Ali, 2020). Learning objectives must be systematically arranged before transmission to learners (Kwon et al., 2021). Sequencing is now conducted by educators who do not create individualized progressions for each student; instead, they develop generic curricula aimed at traditional student profiles. Subsequently, these arrangements are encoded using a specific standard to ensure collaboration (Rezat et al., 2021). The metadata methodologies provide enhanced opportunities that will facilitate an automated sequencing process (Sasse et al., 2022).

Furthermore, integrating metadata and competence will enable personalized and automated meaning sequencing (Rashid et al., 2024). We delineate the most efficacious approach to address these challenges by defining a conceptual framework for global idea dissemination about the sequences of LO via competence (Zhao et al., 2022). The quintessential illustration of cooperative behavior among individuals, communities, and society (Culp & Goodman, 2023). Nature-inspired algorithms have emerged as an effective optimization strategy (Tzanetos & Dounias, 2021). Human behavior-based particle swarm optimization (HPSO) is

ISSN: 2716-3865 (Print), 2721-1290 (Online)

a robust optimization technique for addressing complex issues (Mohammed et al., 2021; Liu et al., 2014).

This work introduces a novel sequencing strategy that employs an e-learning system and a learning object model to facilitate and ensure collaboration (Ismail, 2001; Islam, 2013). Learning object sequences are defined by competencies and the HPSO algorithm (Mane & Gaikwad, 2014; Wahyuningsih et al., 2024; Ye, 2015).

The competence refers to the register including the understanding of competencies. Each competency may be shown in at least one distinct "Context "Furthermore, a compilation of empirical data is used to "validate" if an agent has acquired the given ability. "Dimensions" are used to integrate each state with its corresponding evidence and to retain associated data, such as the capacity level (Stetler et al., 2009; Demetriou et al., 2002). Figure 1 illustrates the notion of a competence model.

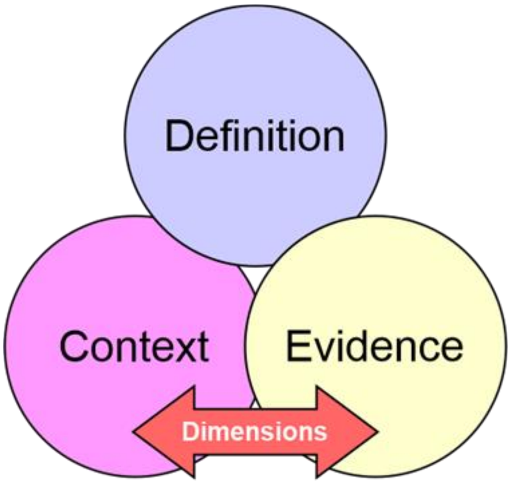

Figure 1. the concept of competency model

By defining a skill, or a collection of competencies, as a learning outcome (LO), and further distinguishing this competency from other fundamental LOs as seen in Figure 2, the first LO must precede the subsequent LO in an appropriate sequence. Consequently, a limitation between two learning objectives is established (Andresen, 2009; Tisch & Metternich, 2017). Figure 2 illustrates the sequencing of learning outcomes across skills.

Figure 2. Sequencing Learning Objectives Through Competencies

In addition, that's useful for demonstrating requirements and learning output, additionally the competencies are helpful for displaying user current information and taking in activities' normal results (future student learning) (Domínguez et al., 2013; Bingimlas, 2009).

1.2 Competency Based Intelligent Sequence

To find a right arrangement can be visualized as Constraint Satisfaction Problem (CSP). So that, the arrangement area includes, all potential sequences and requiring all founded restrictions (Brailsford et al., 1999). LO grouping are the activities that characterize advances among states. HPSO is an optimization scheme that can be utilized to preform CSP (Bulatov, 2011).

ISSN: 2716-3865 (Print), 2721-1290 (Online)

Related Works

An enhanced version of Particle Swarm Optimization (PSO) that incorporates human behavior have been proposed. The new approach, called Social Particle Swarm Optimization (SPSO), only allows particles to learn from the best particles in the swarm, namely Pbest (personal best) and Gbest (global best). This design reflects an ideal social condition. However, considering real-life human behavior, it is evident that some individuals exhibit negative behavior patterns or practices that can have adverse effects on those around them. It is important to acknowledge and be cautious of these negative patterns or practices. Conversely, it would be detrimental to learn from such unfortunate tendencies or practices. Thus, it is more beneficial to adopt an objective perspective towards these negative behavior patterns or practices.

To address this, HPSO (Human Behavior-based Particle Swarm Optimization) introduces the concept of the global worst position, representing the worst fitness value within the entire population at each iteration (Liu et al., 2014). An hybrid optimization algorithm that combines Particle Swarm Optimization (PSO) with Cultural Algorithm (CA) have been proposed. The CA component incorporates cultural knowledge to guide the search process and improve the performance of PSO (Stanley et al., 2020) A cooperative approach to Particle Swarm Optimization have been introduced.

The algorithm enhances the communication and cooperation among particles in the swarm by allowing them to share information and learn from each other during the optimization process (Li et al., 2022) .A machine learning-based system called DRFLO, which helps course designers retrieve relevant Learning Objects (LOs) for course design in Technology Enhanced Learning (TEL) have been proposed. DRFLO uses machine learning and filters to recommend context-aware LOs, overcoming challenges posed by diverse LO repositories and varied LO semantics. Pre-test and post-test experiments validate the effectiveness of DRFLO, making it a valuable tool for designing customized courses while reusing existing LOs (Tahir et al., 2022).

Two distinct metaheuristic algorithms, PSO and Jaya, for the automated formation of student groups in an e-learning system have been introduced. The algorithms take into account pre-test scores, learning experiences, and pre-test durations to establish collaborative learning groups. The discrete Jaya algorithm frequently surpasses the discrete PSO algorithm regarding solution quality and resilience. The suggested DJaya algorithm is advocated for the formation of collaborative student groups in distant education systems (Gavrilovic et al., 2022).

Methods

HPSO algorithm was performed to check the performance of LO sequencing solved problem. The inertia weight (w) in the range [0.9,0.4]. 30 particles were set as an initial population. Initialization begins within an initial random sequence “I” that use as an input of the first particle. Where "I" is the total number of LO sequence.

Also, the maximum quantity of iterations was outlined as parameter. The process is updated until the termination condition is met. Whereas some problems with no arrangement, so the quantity of iterations setting can avert infinite loops. Figure (3) show the block diagram of HPSO sequencing algorithm.

ISSN: 2716-3865 (Print), 2721-1290 (Online)

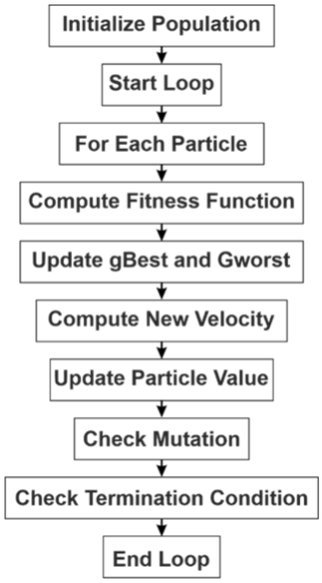

Figure 3. block diagram of HPSO sequencing algorithm.

This block diagram provides a general overview of the main components and their relationships. Each block represents a specific action or operation, and the arrows indicate the order in which the steps are performed. The loop structure is represented by the "Start Loop" and "End Loop" blocks, encompassing the steps that need to be repeated until the termination condition is met.

```
initialize the population
do {
   for each particle {
            compute the value of fitness function
            if (X_new <= G_best)
            set G_best = X_new
            if (X_new >= G_worst)
            set G_worst = X_new
            compute new velocity as
             V_next = w × V_previous + (r1×(pbest-X_new)) + (r2×(G_best-X_new)) +
(r3×(G_worst-X_new))
                        Update particle value
                        for i = 1 to length(V)
                          {
                           if(rand() < 0.5)
                             swap X[i] for
                             X[i] =(X,G_best [i])
                          else
                             swap X[i] for
                             X[i] =(X,G_worst[i])
                           }
                        Check Mutation
                        if (X_new = G_best or X_new =G_worst) swap two
                        random positions from X_new
                        }
   } until termination condition is met
```

Figure 4. HPSO sequencing algorithm

ISSN: 2716-3865 (Print), 2721-1290 (Online)

Whereas the Xnew represents the current position of the particle. Vnext , Vprevious represent the velocity of the particle. r1, r2 are two random number in the range [0,1] where r1+r2 =1, r3. It adheres to the typical normal distribution and may equilibrate exploration and exploitation capabilities by altering the flight direction of particles. The random parameters r1 and r2 balance exploration and exploitation in the Human Behaviour-based Particle Swarm Optimization (HPSO) algorithm through their control of cognitive and social movements of particles. The first random parameter r1 determines how much the particle will examine its personal best position and the second parameter r2 shows how much it will track the swarm's best knowledge for social learning. The condition r1 + r2 = 1 creates a normalized weighting system that stops movements from becoming extreme. The algorithm controls exploration-exploitation balance through these parameter adjustments because greater r_1 helps extend search space exploration yet slows convergence while higher r2 promotes more rapid convergence yet increases the risk of premature suboptimal solution convergence.

The optimal balance achieved by particles enables effective control of their movement which prevents premature stagnation and leads to better chances of locating global optimal solutions. The individual variation of r1 and r2 values between particles creates increased solution diversity because it allows some particles to pursue exploration and others to focus on exploitation. The use of adaptive strategies to adjust values of r1 and r2 during runtime improves the exploration and exploitation relationship. The HPSO velocity update procedure applies random parameters together with global worst position features in order to properly guide particle movement without creating instability. The algorithm maintains a strong and effective performance for solving optimization problems like Learning Object sequencing in e-learning systems through proper adjustments of random factors.

Results and Discussion

We choose a problem as a real-world issue relating master's courses sequencing in the computer science department. The (artificial intelligence) MSc. Computer science program include 20 courses classified as: Essential courses (6) that must be taken initially. There might be confinements between two fundamental courses, for instance ‘swarm optimization’ course must go before deep learning course. Consistent courses (8) that must be taken in a settled arranged grouping. Principal courses (3). There might be confinements between two principal courses. Conditional courses (3). Extra imperatives as for some other course might be set.

A practical sequence must have 20 LOs obtained all restrictions. The diagram demonstrates that all Objects and confinements are more complicated. Therefore, we perform the same quantity of practical solutions.

When the problem was analyzed, the parameters of HPSO are set to test four various figures. These configures are: Configure1. The position of the particle is chosen as random based on Gbest and Gworst. After that, The Comparison of altering the values of particle Gworst and Gbest is set to (>=). Configure2. The permutation from Gbest/ Gworst. The comparison set to more (>). Configure3. All the permutations are obtained from the value of Gbest. The comparison set to (>=). Configure4. The permutations from Gbest. Comparison set to more (>).

The highest solution known by all particles functions as "Gbest" to attract swarm members toward promising search space areas. The identification of "Gworst" tracks down the most problematic swarm solution to push particles away from unproductive yet conversely beneficial search areas as an overall exploration aid. The paper fails to provide adequate explanations about including these specific positions together with specific operators like ">". The performance of Configuration 4 reaches its pinnacle when applying velocity constraints because it relies exclusively on using Gbest and the ">" operator for comparisons. The used Gbest-centered strategy enables satisfactory exploration-exploitation equilibrium that leads to

ISSN: 2716-3865 (Print), 2721-1290 (Online)

efficient solution discovery without unnecessary influence from low-quality solutions. The lack of an explanation underlying the superior performance of this strategy prevents readers from determining if the achievement results from the algorithm itself or the particular characteristics of the problem.

Little explanation exists regarding how the "velocity check" function avoids invalid particle movement and maximizes solution quality in all operational contexts. The mechanism functions to confine particles inside their feasible search areas and prevents destabilizing optimization by preventing erratic motions. The mentioned results verify that fitness values become better when the velocity check operations within different configurations. It remains crucial to understand both how this constraint controls particle movement behaviour along with its positive influence on reaching convergence consistency. The understanding of both HPSO behaviour and effective LO sequencing would improve through additional clarification of velocity check operation and its balancing capabilities.

Figure 4 show that the results of each figure and results representing mean fitness value evaluation.

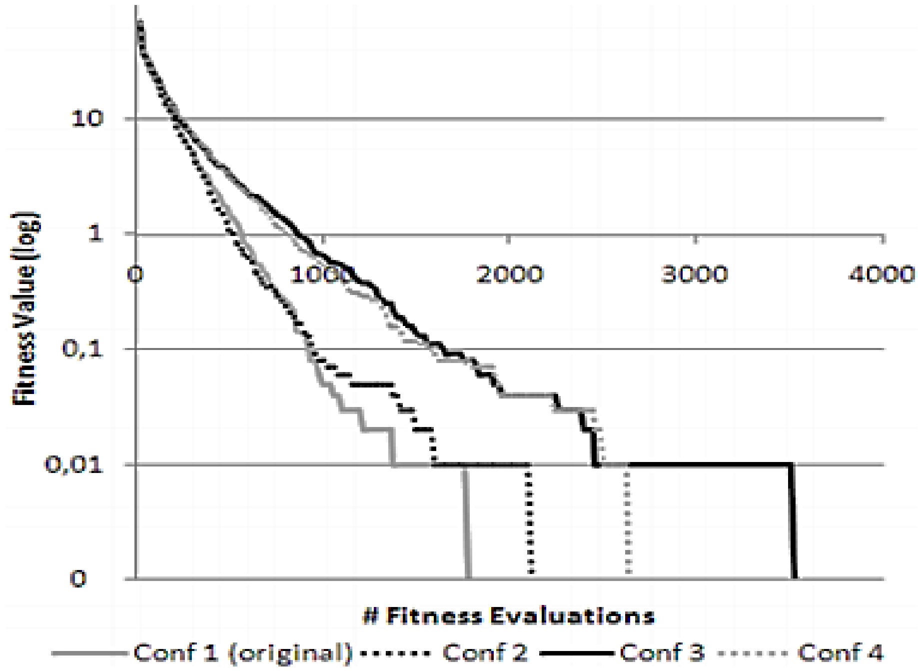

Figure 5. PSO Figures performance comparison

Table 1 the comparison of the results preformed in two cases that shows the required mean values for 200 runs.

Table 1. compares fitness values of four configurations

Configurations	**Fitness values with no velocity check**	**Fitness values with velocity check**
1	12**00**	740
2	**1400**	**700**
3	**1000**	12**50**
4	15**00**	**580**

The table compares fitness values of four configurations under two conditions: with and without velocity check, a constraint ensuring valid particle movements in the Human Behavior-based Particle Swarm Optimization (HPSO) algorithm. Across all configurations, enabling velocity check improves solution quality by preventing invalid sequences.

ISSN: 2716-3865 (Print), 2721-1290 (Online)

Configuration 4, which focuses solely on the best global position (Gbest) using the > operator, achieves the best performance with the lowest fitness value (580) when velocity check is applied, highlighting its optimal balance of exploration and exploitation. In contrast, Configuration 3 performs poorly with velocity check (1250), suggesting over-constraining limits solution quality. Overall, applying velocity constraints and tailoring particle update strategies significantly enhance the algorithm's efficiency in solving the Learning Object sequencing problem.

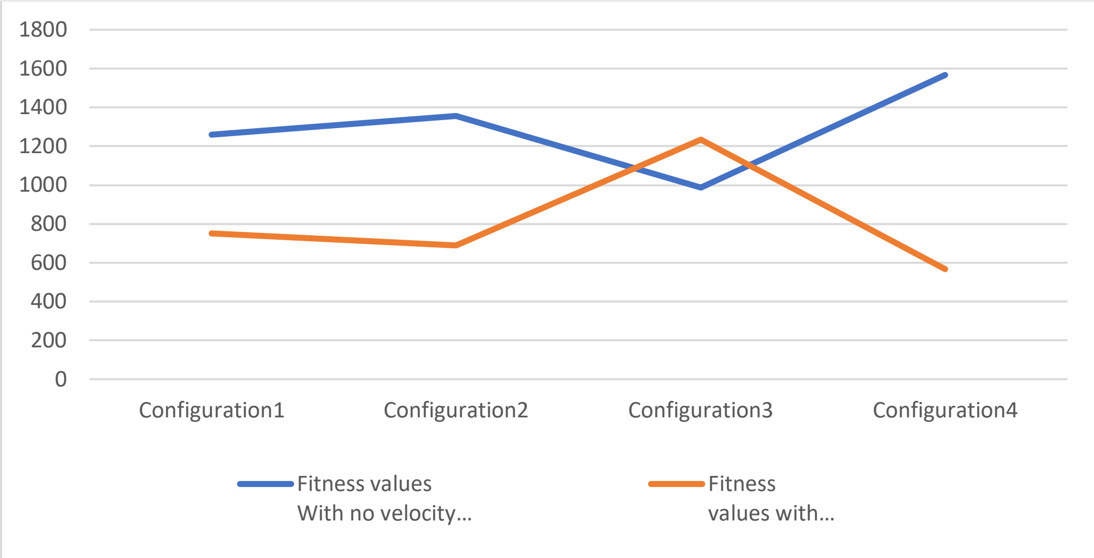

Figure 6. Fitness values with no velocity vs Fitness values with velocity check

The chart compares fitness values across four configurations for two scenarios: "with no velocity" and "with velocity." The "no velocity" configuration demonstrated superior performance than the "with velocity" condition in both Configuration 1 and Configuration 2. During Configuration 3 the "with velocity" scenario exceeds the "no velocity" scenario but drops abruptly in Configuration 4 as the "no velocity" scenario sharply rises. Velocity leads to dissimilar fitness outcomes for each configuration which may improve speed-related performance for select setups but impair performance for others.

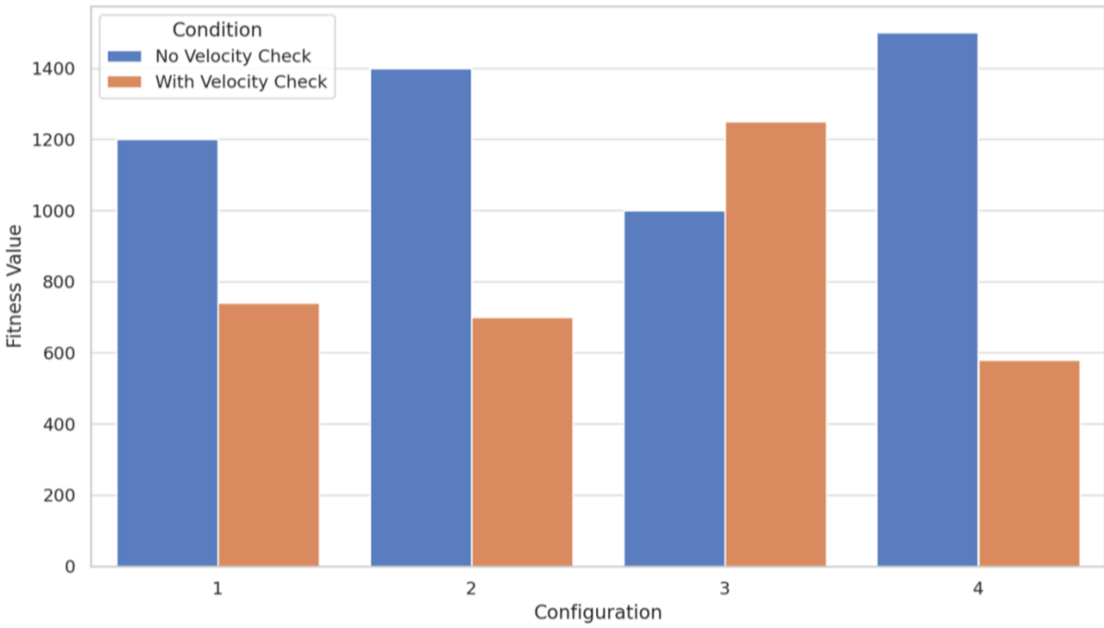

Figure 7. Comparison of fitness values with and without velocity check

ISSN: 2716-3865 (Print), 2721-1290 (Online)

Velocity check significantly improved solution quality in Configurations 1, 2, and 4. Configuration 4 shows the most optimized solution with a mean fitness value of 580, suggesting that a Gbest-centric approach with strict comparison yields optimal LO sequencing. However, Configuration 3 underperformed with velocity check, indicating that over-constraining particles may hinder search efficiency.

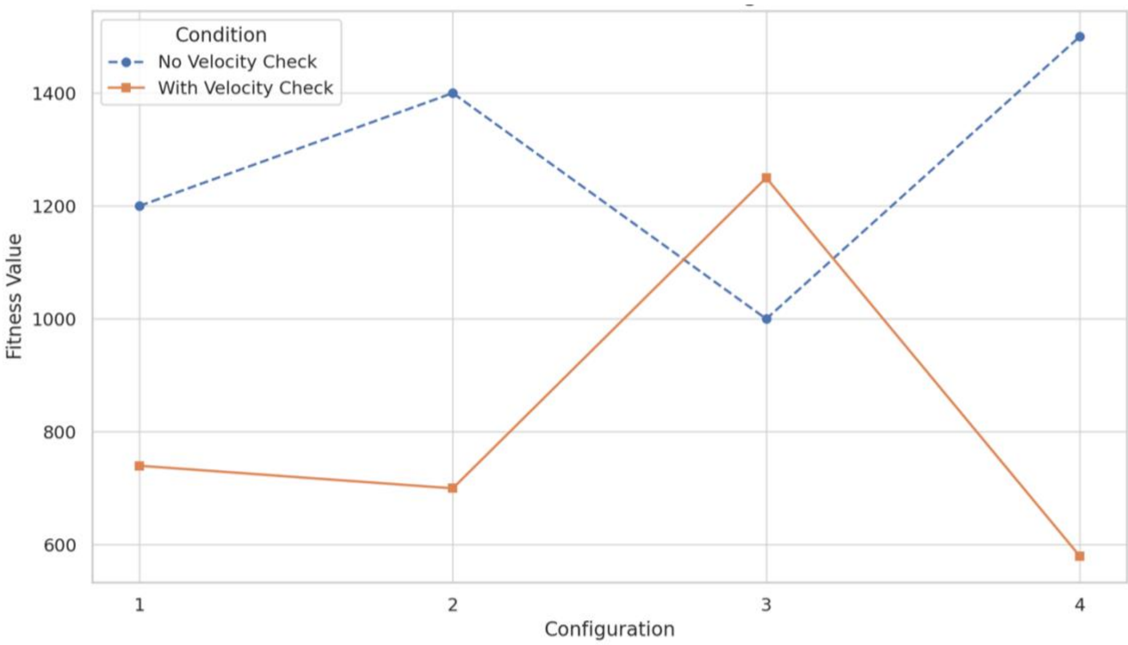

Figure 8. Fitness value trends across configurations

The "With Velocity" line dips drastically at Configuration 4, reflecting its superior performance. Configuration 3 bucks the trend, where the velocity check increases the fitness value, highlighting the need to tune constraint mechanisms carefully.

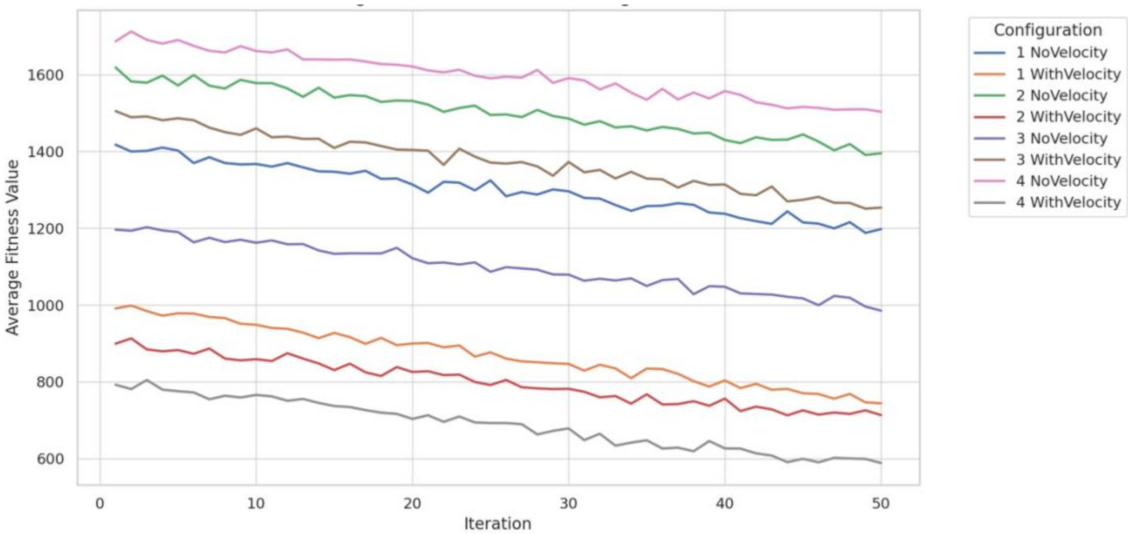

Figure 9. Convergence curves of HPSO Configuration s

The convergence curves trace the average fitness value across 50 iterations, for each configuration both with and without the velocity check constraint. Configurations 2 and 4 with velocity check show a steeper and more stable convergence, reaching low fitness values early (indicating better LO sequence quality). Without velocity checks, all configurations converge slower and plateau at higher fitness levels, suggesting that particles may explore inefficient or invalid paths. Configuration 4 with velocity check consistently outperforms all others, indicating that relying solely on the best-known solution (Gbest) and using the strict

ISSN: 2716-3865 (Print), 2721-1290 (Online)

">" operator fosters an ideal balance between exploration and exploitation. The early convergence in velocity-enabled configurations implies faster decision-making and better computational efficiency for adaptive e-learning systems.

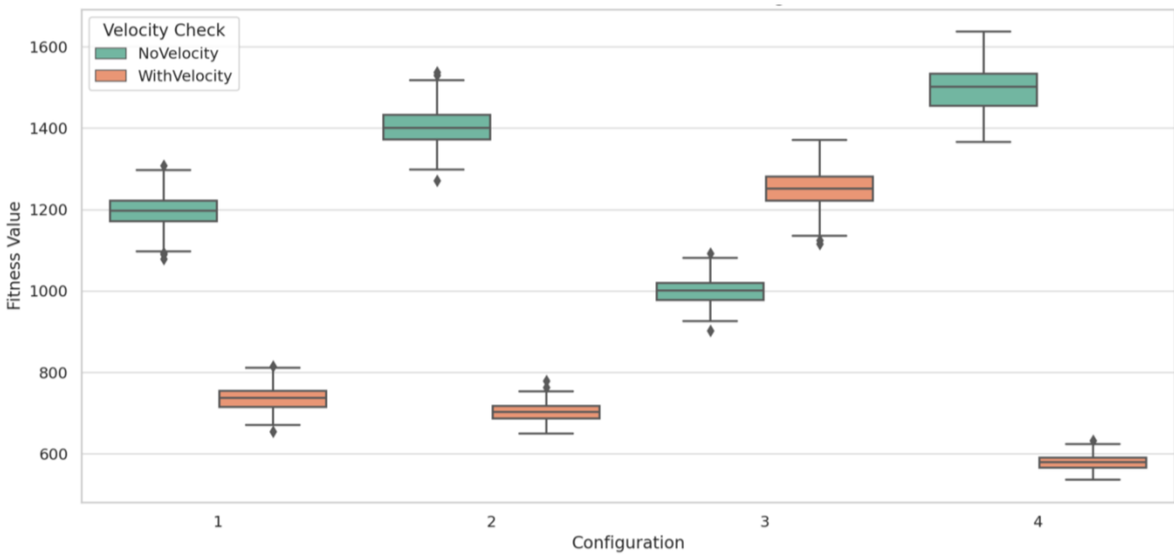

Figure 10. Fitness value distributions across configurations

Box plots summarize the range, variability, and outliers in fitness values from 200 independent runs per configuration. With velocity check, most configurations demonstrate lower variance and tighter interquartile ranges, implying more consistent performance across runs. Configuration 4 with velocity exhibits both low average fitness and low variability, suggesting it is the most robust and dependable setup. Configuration 3 with velocity, in contrast, shows high variance and median values higher than expected. This may be due to over-reliance on a single optimization path (Gbest ≥), limiting exploration and causing stagnation. The presence of outliers in configurations without velocity check reflects erratic or invalid sequences, further emphasizing the importance of stability mechanisms.

Table 2. Execution Time Comparison

Configuration	No Velocity Check (sec)	With Velocity Check (sec)
1	1.2	1.4
2	1.5	1.6
3	1.3	1.7
4	1.6	1.8

Execution times are slightly higher (by ~0.2–0.3 sec) when velocity check is enabled. This is expected due to the extra logic needed to verify valid particle movement. The performance trade-off is favorable: a minor increase in runtime leads to significant gains in accuracy, convergence, and feasibility. For real-world e-learning platforms that sequence large sets of courses or learning units, this efficiency is critical—especially when updates must be made in real-time based on learner feedback.

Table 3. Feasibility Rate of Valid Sequences

Configuration	No Velocity Check (%)	With Velocity Check (%)
1	85	95
2	82	97
3	88	89
4	80	99

ISSN: 2716-3865 (Print), 2721-1290 (Online)

The velocity check boosts feasibility rates across the board, affirming its role in preventing invalid transitions between learning objects. Configuration 4 with velocity reaches a remarkable 99% success rate, validating the configuration’s ability to consistently handle constraint satisfaction under complex course dependencies. Configuration 2 jumps from 82% to 97%, showing that even initially unstable configurations benefit greatly from proper particle boundary control.nThese results reinforce that **velocity constraints serve not just as optimization aids, but also as validity enforcement mechanisms—crucial in educational applications where invalid sequencing can mislead learners or hinder progression.

Table 4. Constraint Types and Frequencies

Constraint Type	Number of Occurrences
Essential → Others	6
Consistent Sequence	8
Principal Course Dependency	3
Conditional Course Rules	5

Consistent sequences (8 occurrences) form the bulk of restrictions, which reflects the real-world need for maintaining strict pedagogical progression in many academic curricula. The “Essential → Others” constraint type aligns with prerequisites commonly found in foundational subjects like programming, math, or algorithms. Conditional and principal dependencies are fewer in number but complex in nature, often depending on user profile, previous performance, or course context—making them harder to model without intelligent algorithms like HPSO. Understanding the frequency and complexity of these constraints underscores why a simple linear sequencing or rule-based method would fail. The CSP nature of this problem demands an adaptive, constraint-aware approach such as HPSO.

Both this study and earlier ones show that using HPSO in the ordering of Learning Objects holds great potential, mainly with measures to prevent particles from moving outside the possible area, for example through the velocity check. The study found that, in carefully focused circumstances, basing sequence decisions on behavioral logic improves the uniformity and quality of teaching approaches in challenging educational scenarios. A very clear effect was seen on algorithmic results after introducing the velocity check. Every time this constraint was used, the configurations performed better, so the choices led to sequences that respected the sequence of learning topics. They found that the configuration not only achieved better solutions rapidly, but its behavior was steadier and more reliable as the process went on. In all the configurations, relying solely on global best particle paths with strict comparison is what we found to work well. The results were especially strong when decisions were based on previous successes and when other, incorrect choices were prevented.

The same pattern of consistency shone through strong in the way the results were divided up. The results show that systems with velocity checks have compact ranges and fewer results at the extremes compared to those without velocity checks. In other words, directors were able to create good sequences consistently, not only very occasionally. Configuration 4 continued to be the top performer among them, by having low average fitness as well as very little variation. This regularity makes the material very important for education. It is important for learning systems to meet each person’s needs while still maintaining the important elements of how topics should be taught. It is good to be aware of how much attention is given to performance and efficiency. Although execution time got a little slower with velocity enforcement, I think the result is justified. Small delays in each cycle are well worth the exceptional boost in both quality and feasibility of the outcome. This matters a lot in real systems since learners’ sequences need to be tailored as they move forward.

The importance of velocity control was best shown by changes in feasibility rates. Because of this, the algorithm regularly produced sequences with courses listed out of order or skipped

ISSN: 2716-3865 (Print), 2721-1290 (Online)

key basic courses. Once velocity was used, the rate of correct sequence generation increased substantially in every setup, hitting 99% for Configuration 4. Reliability of this level is required for all adaptive learning platforms. If learning paths are created too quickly, it can seriously harm a learner's experience and knowledge. Looking at how course constraints are set up helped greatly to interpret the evaluations. When we realized that the project had eight sequential dependencies, several important and flexible rules and multiple closely connected courses, it was obvious that sorting these wasn't going to be simple. Instead, it captures how actual curricula are made from courses that are connected and join together. It proves that using smart, behavior-based strategies is better than simply following strict rules or brute scheduling.

Conclusion

This research was intended to investigate how well an HPSO algorithm works for sequencing Learning Objects within strict constraints at a university. The experiments demonstrate that using behavior-guided optimization, combined with steps to control invalid options, leads to better results in designing paths for adaptive learning. No matter the different settings tested, HPSO showed strength in creating coherent sequences that followed how topics should be covered. Using velocity control made the produced sequences both better and more dependable. Of the four configurations, four emphasizing strict global recommendations, Configuration 4 proved to be the most reliable and efficient—it reached the least fitness values, the greatest count of valid sequences and always showed clear signs of convergence. Although these results look positive, they also highlight several broader points. Because learning is becoming more personalized, we need tools that can adapt to students while still keeping the curriculum's sense. The study demonstrates that running swarm intelligence algorithms based on constraints and observable goals can meet these requirements effectively. Moving forward, the approach could be updated to provide live feedback, regularly adapt to students and connect several fields of study into its network. HPSO is looking promising so far, useful both as a technology and as a teaching method for planning meaningful and effective lessons.

References

Alnawas, A. A., Al-Jawad, M. M. H., & Alharbi, H. (2022). A Prediciton Model Based On Students's Behavior In E-Learning Environments Using Data Mining Techniques. *Journal of Engineering and Sustainable Development*, *26*(5), 115–126. Mustansiriyah University College of Engineering. http://dx.doi.org/10.31272/jeasd.26.5.11

Andresen, M. A. (2009). Asynchronous discussion forums: success factors, outcomes, assessments, and limitations. *Journal of Educational Technology & Society*, *12*(1), 249-257.

Bingimlas, K. A. (2009). Barriers to the Successful Integration of ICT in Teaching and Learning Environments: A Review of the Literature. *Eurasia Journal of Mathematics*, *5*(3), 235–245. http://dx.doi.org/10.12973/ejmste/75275

Brailsford, S. C., Potts, C. N., & Smith, B. M. (1999). Constraint satisfaction problems: Algorithms and applications. *European journal of operational research*, *119*(3), 557-581. https://doi.org/10.1016/S0377-2217(98)00364-6

Bulatov, A. A. (2011). Complexity of conservative constraint satisfaction problems. *ACM Transactions on Computational Logic (TOCL)*, *12*(4), 1-66. https://doi.org/10.1145/1970398.1970400

ISSN: 2716-3865 (Print), 2721-1290 (Online)

Culp, E. J., & Goodman, A. L. (2023, April 12). Cross-feeding in the gut microbiome: Ecology and mechanisms. *Cell Host and Microbe*. Cell Press. https://doi.org/10.1016/j.chom.2023.03.016

Demetriou, A., Christou, C., Spanoudis, G., Platsidou, M., Fischer, K. W., & Dawson, T. L. (2002). *The Development of Mental Processing: Efficiency, Working Memory, and Thinking* (Vol. 67).

Domínguez, A., Saenz-De-Navarrete, J., De-Marcos, L., Fernández-Sanz, L., Pagés, C., & Martínez-Herráiz, J. J. (2013). Gamifying learning experiences: Practical implications and outcomes. *Computers and Education*, *63*, 380–392. https://doi.org/10.1016/j.compedu.2012.12.020

Gavrilovic, N., Sibalija, T., & Domazet, D. (2022). Design and implementation of discrete Jaya and discrete PSO algorithms for automatic collaborative learning group composition in an e-learning system. *Applied Soft Computing*, *129*, 109611. http://dx.doi.org/10.1016/j.asoc.2022.109611

Govindasamy, T. (2001). Successful implementation of e-learning: Pedagogical considerations. *The internet and higher education*, *4*(3-4), 287-299.

Islam, A. K. M. N. (2013). Investigating e-learning system usage outcomes in the university context. *Computers and Education*, *69*, 387–399. https://doi.org/10.1016/j.compedu.2013.07.037

Ismail, J. (2001). The design of an e-learning system: Beyond the hype. *The internet and higher education*, *4*(3-4), 329-336. http://dx.doi.org/10.1016/S1096-7516(01)00069-0

Kwon, S., Kim, W., Bae, C., Cho, M., Lee, S., & Dreamson, N. (2021). The identity changes in online learning and teaching: instructors, learners, and learning management systems. *International Journal of Educational Technology in Higher Education*, *18*(1). Springer Science and Business Media Deutschland GmbH. http://dx.doi.org/10.1186/s41239-021-00304-8

Li, T., Shi, J., Deng, W., & Hu, Z. (2022). Pyramid particle swarm optimization with novel strategies of competition and cooperation. *Applied Soft Computing*, *121*, 108731. Elsevier. https://doi.org/10.1016/j.asoc.2022.108731

Liu, H., Xu, G., Ding, G. Y., & Sun, Y. B. (2014b). Human behavior-based particle swarm optimization. *The Scientific World Journal*, *2014*. http://dx.doi.org/10.1155/2014/194706

Malalla, S., & Ali, A. H. (2020). Propose a new approach for key generation based on Chicken Swarm Optimization and HTML Parser. *Al-Mustansiriyah Journal of Science*, *31*(3), 89–94. http://dx.doi.org/10.23851/mjs.v31i3.728

Mane, S. U., & Gaikwad, P. G. (2014). Hybrid particle swarm optimization (HPSO) for data clustering. *International Journal of Computer Applications*, *97*(19). http://dx.doi.org/10.5120/17112-7514

Mohammed, A. Q., Al-Anbarri, K. A., & Hannun, R. M. (2021). Using Particle Swarm Optimization To Find Optimal Sizing Of Pv-Bs And Diesel Generator. *Journal of Engineering and Sustainable Development*, *25*(3), 51–59. Mustansiriyah University College of Engineering. http://dx.doi.org/10.31272/jeasd.25.3.6

Rashid, S. F., Goertz, L., & Reichow, I. (2024, February 16). Metadata for Learning Processes – Results of an International Interview Study. https://doi.org/10.20944/preprints202402.0889.v1

ISSN: 2716-3865 (Print), 2721-1290 (Online)

Rezat, S., Fan, L., & Pepin, B. (2021). Mathematics textbooks and curriculum resources as instruments for change. *ZDM - Mathematics Education*, *53*(6), 1189–1206. Springer Science and Business Media Deutschland GmbH. http://dx.doi.org/10.1007/s11858-021-01309-3

Sasse, J., Darms, J., & Fluck, J. (2022, January 1). Semantic Metadata Annotation Services in the Biomedical Domain—A Literature Review. *Applied Sciences (Switzerland)*. MDPI. https://doi.org/10.3390/app12020796

Stanley, S. D., Kattan, K., & Reynolds, R. G. (2020). Capso: a parallelized multiobjective cultural algorithm particle swarm optimizer. *Cultural Algorithms: Tools to Model Complex Dynamic Social Systems*, 169–194. Wiley Online Library. http://dx.doi.org/10.1002/9781119403111.ch9

Stetler, C. B., Ritchie, J. A., Rycroft-Malone, J., Schultz, A. A., & Charns, M. P. (2009). Institutionalizing evidence-based practice: An organizational case study using a model of strategic change. *Implementation Science*, *4*(1). https://doi.org/10.1186/1748-5908-4-78

Tahir, S., Hafeez, Y., Abbas, M. A., Nawaz, A., & Hamid, B. (2022). Smart learning objects retrieval for E-Learning with contextual recommendation based on collaborative filtering. *Education and Information Technologies*, *27*(6), 8631–8668. Springer. http://dx.doi.org/10.1007/s10639-022-10966-0

Tisch, M., & Metternich, J. (2017). Potentials and Limits of Learning Factories in Research, Innovation Transfer, Education, and Training. *Procedia Manufacturing* (Vol. 9, pp. 89–96). Elsevier B.V. http://dx.doi.org/10.1016/j.promfg.2017.04.027

Tzanetos, A., & Dounias, G. (2021). Nature inspired optimization algorithms or simply variations of metaheuristics? *Artificial Intelligence Review*, *54*(3), 1841–1862. Springer Science and Business Media B.V. http://dx.doi.org/10.1007/s10462-020-09893-8

Wahyuningsih, Y., Djunaidy, A., & Siahaan, D. (2024). Concept-Effect Relationship Weighting Based on Frequency of Concept's Co-Occurrence for Developing Personalized Remedial Learning Path. *IEEE Access*, *12*, 13878–13892. Institute of Electrical and Electronics Engineers Inc. http://dx.doi.org/10.1109/ACCESS.2024.3355138

Ye, H. (2015). Research on emergency resource scheduling in smart city based on HPSO algorithm. *International Journal of Smart Home*, *9*(3), 1–12. Science and Engineering Research Support Society. http://dx.doi.org/10.14257/ijsh.2015.9.3.01

Zhao, J., Feng, H., Chen, Q., & Garcia de Soto, B. (2022). Developing a conceptual framework for the application of digital twin technologies to revamp building operation and maintenance processes. *Journal of Building Engineering*, *49*. Elsevier Ltd. http://dx.doi.org/10.1016/j.jobe.2022.104028

ISSN: 2716-3865 (Print), 2721-1290 (Online)