2026 ASEE ANNUAL CONFERENCE June 21–24, 2026 | Charlotte Convention Center, NC
Host Sponsor UNC Charlotte
Paper ID #50786

Teaching Computational Intelligence Course by a Spiral Curriculum and CDIO-based Hybrid Pedagogical Approach

Dr. Tingjun Lei, University of North Dakota

Dr. Tingjun Lei is currently an Assistant Professor in the School of Electrical Engineering and Computer Science at the University of North Dakota. He received his Ph.D. degree in electrical and computer engineering with the Department of Electrical and Computer Engineering, Mississippi State University, Mississippi State, MS, USA., in 2023, his M.S. degree in electrical and computer engineering from the New York Institute of Technology, Old Westbury, NY, USA, in 2016, and the B.S. degree in transportation engineering from Shanghai Maritime University, Shanghai, China, in 2014. He developed and taught the CSCI 242 Algorithms and Data Structure, EE 457 Robotics Fundamentals, EECS 590 Data Science in Robotics and EE 752 Introduction to Autonomous Systems at UND, providing hands-on experience and mentoring to undergraduate and graduate students, fostering a deep understanding of basic algorithms, data analysis, autonomous robot systems and applications. Dr. Lei received the Best Paper Awards from Biomimetic Intelligence and Robotics (Elsevier), 2025 Cyber Awareness and Research Symposium 2026 (CARS 2025), 2022 International Conference on Swarm Intelligence (ICSI 2022) and the 2024 American Society for Engineering Education (ASEE ECE Division). He also received Best of Track Paper Award from Integrated Communications Navigation and Surveillance (ICNS 2026) conference, and Honorable Mentions Awards from the International Joint Conference on Neural Networks (IJCNN 2025), IEEE International Conference on Robotics and Biomimetics (IEEE ROBIO 2025), IEEE International Midwest Symposium on Circuits and Systems (IEEE MWSCAS 2025). Dr. Lei serves as an Editorial Board Member for Intelligence and Robotics, Robot Learning, IET Cyber-Systems and Robotics, Designs, and Biomimetic Intelligence and Robotics. He has also actively contributed as a Technical Program Committee (TPC) member for numerous international conferences, including IEEE CEC, IEEE IJCNN, ICSI, and PRIS. He was awarded the Early Career Scholars Award by the University of North Dakota. His two papers have been selected and featured as cover articles on Intelligence and Robotics. His research spans multiple interdisciplinary areas, including bio-inspired artificial intelligence (AI), robotics and autonomous systems, optimization and evolutionary computation, human-autonomy teaming (HAT), intelligent transportation systems, and applied machine learning.

Timothy Sellers, Mississippi State University

Timothy Sellers (Student Member, IEEE) received the B.S. degree in Robotics and Automation Technology and Applied Science in Electro-Mechanical Engineering from the Alcorn State University, Lorman, MS, USA in 2020. He is currently pursuing a Ph.D. degree in the Department of Electrical and Computer Engineering at Mississippi State University, Mississippi State, MS, USA. His research interests include robotics, autonomous systems, navigation, nature-inspired intelligence, and machine learning.

Dr. Chaomin Luo, Mississippi State University

Dr. Chaomin Luo received his Ph.D. degree in electrical and computer engineering from the Department of Electrical and Computer Engineering at the University of Waterloo, Canada. He also earned an M.Sc. degree in engineering systems and computing from the University of Guelph, Canada, and a B.Sc. in electrical engineering from Southeast University. Currently, he is a Professor in the Department of Electrical and Computer Engineering at Mississippi State University. His research interests include engineering education, robotics, control, automation, and robotics education. He is Associate Editor in 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2019). He is Tutorials Co-Chair in the 2020 IEEE Symposium Series on Computational Intelligence. Dr. Luo was the recipient of the Best Paper Awards in ASEE2024, Biomimetic Intelligence and Robotics (Journal) 2024, IEEE International Conference on Information and Automation, International Conference on Swarm Intelligence, and SWORD Conference. Dr. Luo is active nationally and internationally in his research field. He was the Program Co-Chair in 2018 IEEE International Conference on Information and Automation (IEEE-ICIA'2018). He was the Plenary Session Co-Chair in the 2021 and 2019 International Conference

on Swarm Intelligence, and he was the Invited Session Co-Chair in the 2017 International Conference on Swarm Intelligence. He was the General Co-Chair of the 1st IEEE International Workshop on Computational Intelligence in Smart Technologies (IEEE-CIST 2015), and Journal Special Issues Chair, IEEE 2016 International Conference on Smart Technologies (IEEE-SmarTech), Cleveland, OH, USA. He was Chair and Vice Chair of IEEE SEM - Computational Intelligence Chapter and was a Chair of IEEE SEM - Computational Intelligence Chapter and Chair of Education Committee of IEEE SEM. He has organized and chaired several special sessions on topics of Intelligent Vehicle Systems and Bio-inspired Intelligence in reputed international conferences such as IJCNN, IEEE-SSCI, IEEE-CEC, IEEE-CASE, and IEEE-Fuzzy, etc. He has extensively published in reputed journals and conference proceedings, such as IEEE Transactions on Industrial Electronics, IEEE Transactions on Neural Networks and Learning Systems, IEEE Transactions on SMC, IEEE Transactions on Cybernetics, IEEE-ICRA, and IEEE-IROS, etc. Dr. Luo serves as Associate Editor of IEEE Transactions on Cognitive and Developmental Systems, International Journal of Robotics and Automation, Biomimetic Intelligence and Robotics, and Intelligence and Robotics.

Prof. Zhuming Bi P.E., Purdue University Fort Wayne

Zhuming Bi (Senior Member, IEEE) received the Ph.D. degree from the Harbin Institute of Technology, Harbin, China, in 1994, and the Ph.D. degree from the University of Saskatchewan, Saskatoon, SK, Canada, in 2002. He has international work experience in Mainland China, Hong Kong, Singapore, Canada, UK, Finland, and USA. He is currently a professor of Mechanical Engineering with Purdue University Fort Wayne, Fort Wayne, IN, USA. His current research interests include robotics, mechatronics, Internet of Things (IoT), digital manufacturing, automatic robotic processing, and enterprise information systems. He has published 6 research books and over 180 journal publications in these fields.

Dr. Umar Iqbal, Illinois State University

Dr. Iqbal is an Associate Professor at Electrical Engineering, Illinois State University. He worked in the areas of Multi-Sensor Integration, Navigation Systems, Robotics, Control, and Educational Research. Dr. Iqbal has designed and taught over 25 courses, spanning topics from robotics and control systems to circuits and design. His research, including blended learning and COVID-19's impact, shapes modern engineering education.

Teaching Computational Intelligence Course by a Spiral Curriculum and CDIO-based Hybrid Pedagogical Approach

Tingjun Lei[1], Timothy Sellers[2], Chaomin Luo[2], Zhuming Bi[3], and Umar Iqbal[4]

[1]School of Electrical Engineering and Computer Science, University of North Dakota

[2]Department of Electrical and Computer Engineering, Mississippi State University

[3]Department of Civil and Mechanical Engineering, Purdue University Fort Wayne

[4]Department of Electrical Engineering, Illinois State University

Abstract

This paper describes the design, implementation, and assessment of a Computational Intelligence (CI) course in Electrical and Computer Engineering (ECE) using a hybrid pedagogical approach that integrates the Conceive-Design-Implement-Operate (CDIO) framework with a spiral curriculum (SC). The course is designed to strengthen students' ability to apply CI theory to engineering practice while developing professional competencies aligned with ABET Student Outcomes. Core topics include evolutionary computation, swarm intelligence, bio-inspired neural networks, fuzzy systems, and nature-inspired optimization. CDIO provides a structured, project-based learning environment in which students address real-world CI problems, while the SC reinforces foundational concepts through repeated exposure at increasing levels of complexity. Team-based projects guide students through problem formulation, algorithm design, computational implementation, and performance evaluation, emphasizing trade-offs such as accuracy versus efficiency and adaptability versus convergence. Assessment methods include simulations, technical reports, presentations, reflective journals, examinations, and ABET-aligned rubrics targeting problem solving, design, experimentation, teamwork, communication, and ethical responsibility. Comparative analysis of course outcomes, student surveys, and evaluations across multiple offerings indicates improved engagement, conceptual retention, and system-level thinking. The results suggest that the CDIO-SC hybrid pedagogy effectively supports ABET-aligned learning outcomes and provides a scalable instructional model for CI education in modern ECE curricula.

1 Introduction

Computational Intelligence (CI) has become a foundational component of modern Electrical and Computer Engineering (ECE), enabling practical solutions in optimization, robotics, control, embedded intelligence, and data-driven decision-making [1–4]. As CI techniques continue to expand in scope and engineering relevance, many ECE programs have introduced CI courses to prepare students to translate biologically inspired and nature-inspired algorithms into deployable systems [5–10]. However, teaching CI effectively remains challenging. A typical CI syllabus spans heterogeneous paradigm, evolutionary computation, swarm intelligence, fuzzy systems, and bio-inspired neural networks, each with distinct modeling assumptions, implementation patterns, and evaluation criteria. Students often experience CI as a collection of disconnected "algorithm blocks," which makes it difficult to develop transferable understanding, system-level thinking, and the ability to justify engineering trade-offs.

Engineering education research has demonstrated that student learning improves when instruction moves beyond passive content delivery and emphasizes active construction, iteration, and authentic practice. Interactive learning activities can yield higher learning gains than purely passive exposure [11–16]. Inquiry-based and virtual-lab experiences have been shown to strengthen conceptual understanding and procedural skills, particularly when learners are guided to connect observations with underlying principles [17]. Project-based learning (PBL) has likewise been widely adopted to bridge theory and engineering practice and to cultivate professional competencies, including design reasoning, communication, and teamwork [18–20]. In addition, flipped learning can improve engagement and learning outcomes by reallocating class time toward problem solving and discussion, though sustained benefits depend on deliberate scaffolding and repeated use [21–23]. Despite these advances, CI instruction frequently applies such methods in isolation (*e.g.*, a single capstone project or a set of unconnected labs), which may not be sufficient to support cumulative learning across diverse CI paradigms or to maintain explicit alignment with ABET Student Outcomes.

This paper addresses the above gap by introducing and evaluating a hybrid pedagogical approach for a CI course that integrates the Conceive-Design-Implement-Operate (CDIO) framework with a spiral curriculum (SC) structure. To support cumulative learning across diverse CI paradigms and strengthen the link between theory and engineering practice, we adopt a hybrid pedagogy that combines the CDIO framework with a spiral curriculum (SC). CDIO provides the organizing backbone that guides students from problem conception and algorithm design to implementation and system-level evaluation [24]. The SC reinforces shared CI principles (e.g., optimization, learning, robustness, and uncertainty) through repeated exposure at increasing levels of technical depth, while milestone-driven deliverables and guided reflections promote self-regulation and deeper learning [25–28].

The proposed approach is implemented in a 15-week ECE CI course covering evolutionary computation, swarm intelligence, fuzzy systems, and bio-inspired neural networks. Team-based projects guide students through CI problem definition, algorithmic modeling, implementation in MATLAB/Python, and performance evaluation with explicit trade-off analysis (*e.g.*, accuracy versus computational cost, exploration versus exploitation, and adaptability versus convergence). Assessments include simulations, technical reports, presentations, examinations, reflective journals, and ABET-aligned rubrics. The contributions of this paper are threefold: (i) A scalable course architecture is presented that fuses CDIO stages with spiral reinforcement across CI paradigms; (ii) concrete examples of CDIO-SC integration are provided through representative CI projects (*e.g.*, PSO-based path planning and neuro-fuzzy control); and (iii) an assessment strategy is reported that maps course activities to ABET Student Outcomes and uses multiple evidence sources to evaluate student learning and professional skill development.

The remainder of this paper is organized as follows. Section 2 describes the CDIO-SC hybrid pedagogy and how it is operationalized through milestone-driven projects. Section 3 details the course structure, weekly topic sequencing, and representative learning activities. Section 4 presents the assessment methodology and summarizes evaluation results and observed learning outcomes, followed by a conclusion in Section 5.

2 The Hybrid Pedagogical Approach of CDIO and SC

This section presents the pedagogical approach that integrates the CDIO with the SC to support both engineering practice and progressive conceptual reinforcement in CI education.

2.1 Pedagogical Approach Based on the CDIO Framework

The pedagogical approach adopted in this course is grounded in the CDIO framework, which provides a structured yet flexible model for integrating theory, practice, and professional skills in CI education. CDIO is used as the primary organizational backbone of the course, guiding students through the complete engineering lifecycle of CI-based systems while emphasizing real-world relevance and active learning. In the Conceive phase, students identify meaningful engineering problems that can be addressed using CI techniques, such as optimization under uncertainty, adaptive decision-making, or intelligent navigation. They analyze problem requirements, constraints, and application contexts, fostering systems thinking and problem formulation skills (see Figure 1). In the Design phase, students develop algorithmic solutions using evolutionary computation, fuzzy systems, swarm intelligence, or bio-inspired neural networks. This stage emphasizes modeling choices, parameter selection, and trade-off analysis, including accuracy versus computational efficiency and exploration versus exploitation. During the Implement phase, students translate their designs into executable computational models using simulation tools such as MATLAB or Python, gaining hands-on experience in algorithm development, debugging, and performance tuning. The Operate phase focuses on evaluation and validation, where students test their CI systems on benchmark datasets or application-driven scenarios, assess robustness and convergence behavior, and iteratively refine their solutions based on quantitative results. To connect CI topics across the semester, the spiral reinforcement mechanism that complements the CDIO backbone is described next.

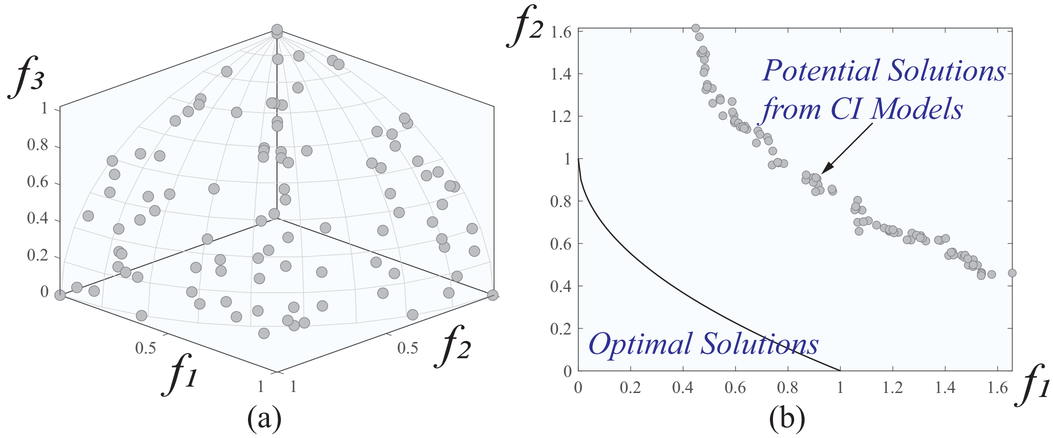

Figure 1: The hybrid pedagogical approach for the typical CI multi-objective optimization. The grey dots in the figure are potential solutions from CI algorithms. (a) Three criteria. (b) Two criteria. [29]

2.2 *Fusion of CDIO and SC Pedagogical Approaches*

To reinforce practical competency and system-level reasoning, the course adopts a SC structure that repeatedly engages students with core CI concepts across multiple engineering contexts. Rather than covering each CI technique in isolation, foundational ideas such as optimization, learning, robustness, and adaptability are introduced early and revisited through evolutionary computation, fuzzy systems, swarm intelligence, and bio-inspired neural networks. Each iteration increases in technical depth and implementation complexity, allowing students to strengthen their understanding through hands-on application. For example, optimization is first explored using basic search methods, then extended through genetic algorithms and swarm-based approaches and later integrated into hybrid neuro-fuzzy or evolutionary systems. This spiral structure mirrors real-world engineering practice, where solutions are refined iteratively as system requirements evolve. Assignments and projects are designed to build upon prior work, encouraging students to reuse, modify, and improve earlier designs while evaluating performance trade-offs such as accuracy, convergence speed, and computational cost. The SC also supports students with varied preparation levels by enabling gradual skill development through repeated exposure and application. When combined with CDIO-based projects, this approach promotes continuity between theory and practice, enhances problem-solving efficiency, and prepares students to apply CI techniques effectively in realistic engineering scenarios.

The CI course is structured around the CDIO cycle, with CI-centered projects serving as the primary learning vehicle. A representative example used throughout the course is the application of Particle Swarm Optimization (PSO) to robotic path planning problems (see Figure 2).

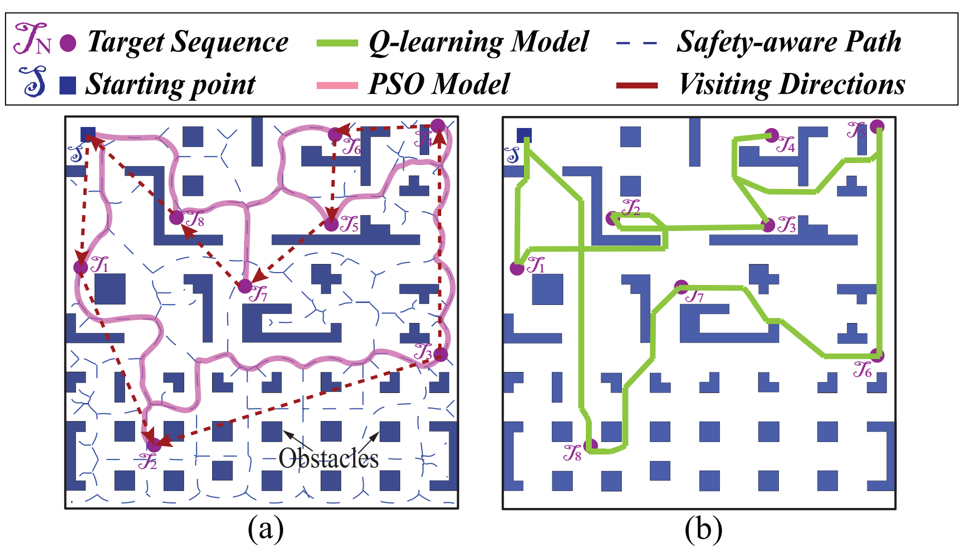

Figure 2: Illustration of the path created from different CI models [30–33]. The point order is illustrated by the violet arrows while the orange path represent the robot path. Safety-aware roads are depicted by the blue dashed lines. The waypoints are illustrated by the violet circles. (a) It depicts the path created by PSO model by the pink lines. (b) It depicts the path created by Q-learning model by the green lines.

- o *Conceive*: Students begin by identifying engineering challenges such as autonomous robot navigation in cluttered or dynamic environments, where traditional deterministic planners may struggle. At this stage, students define performance objectives, constraints, and evaluation criteria, including path length, collision avoidance, convergence speed, and computational cost, while considering practical deployment contexts in robotics and embedded systems.

- o *Design*: Based on the problem formulation, students design PSO-based path planning algorithms by modeling particle representations, fitness functions, and update rules. Emphasis is placed on algorithmic parameter selection, swarm topology, and trade-offs between exploration and exploitation. Mathematical modeling and simulation are used to validate design choices and compare PSO with classical or alternative CI-based planners.
- o *Implement*: Students implement their PSO algorithms in simulation environments such as MATLAB or Python, integrating sensor models and environmental constraints. Through iterative testing and debugging, they gain hands-on experience translating CI theory into executable systems.
- o *Operate*: Finally, students evaluate algorithm performance using benchmark environments or robotics scenarios, analyze robustness and convergence behavior, and refine designs through iterative optimization. Learning outcomes are assessed through simulation results, technical reports, presentations, and reflective analyses. Evaluation criteria emphasize not only technical correctness and innovation, but also teamwork, communication, and the ability to integrate CI theory with engineering practice.

A second representative example focuses on neuro-fuzzy control for adaptive robotic motion or process control. In this CI class, fuzzy systems were taught using a Takagi-Sugeno fuzzy logic model applied to a real-world scenario involving a recharging station. When an autonomous robot approaches the vicinity of the station, it must be guided to align correctly for docking. This is achieved through a novel heuristic error-control approach that adjusts both the angle and distance of the robot, ensuring it approaches the recharging station at the proper orientation. Figure 3 illustrates the fuzzy logic model used to control the robot's alignment angles.

- o *Conceive*: Students consider control problems involving uncertainty, nonlinearity, or incomplete models, such as velocity control of a mobile robot or temperature regulation in an embedded system.
- o *Design*: They develop neuro-fuzzy architectures that integrate fuzzy rule-based reasoning with neural network learning to achieve adaptive parameter tuning.
- o *Implement*: Students implement neuro-fuzzy controllers in simulation, training the system using input-output data and evaluating stability and responsiveness.
- o *Operate*: Controller performance is assessed under varying conditions, emphasizing robustness, adaptability, and real-time feasibility. Across both examples, learning outcomes are evaluated through simulations, technical reports, presentations, and reflective analyses, with assessment criteria addressing technical accuracy, innovation, teamwork, communication, and the integration of CI theory with engineering practice.

The PSO-based path planning and neuro-fuzzy control examples illustrate how the fusion of the CDIO framework with a SC is operationalized in the CI course. In both cases, students repeatedly engage with core CI concepts such as optimization, learning, adaptation, and uncertainty across

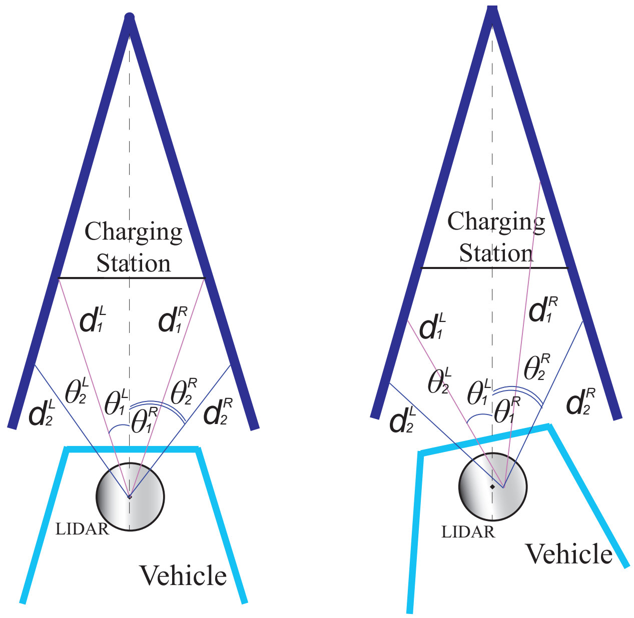

Figure 3: Illustration of neuro-fuzzy logic model used to control the robot's alignment angles [34].

multiple CDIO stages. PSO-based path planning reinforces optimization and collective intelligence principles, while neuro-fuzzy control revisits learning and reasoning under uncertainty through adaptive rule tuning. These concepts are first introduced at a conceptual level, then revisited with increasing algorithmic depth and implementation complexity as students' progress through design, implementation, and evaluation phases shown in Table 1. By encountering shared principles across distinct CI paradigms and application contexts, students develop transferable knowledge and system-level understanding. The spiral structure ensures continuity and reinforcement, while the CDIO framework provides a practical engineering lifecycle for applying and refining CI solutions. Together, these examples demonstrate how the integrated CDIO-SC pedagogy promotes cumulative learning, strengthens the connection between theory and practice, and prepares students to apply CI techniques effectively in realistic engineering scenarios.

Table 1: Neuro-fuzzy control example: Integrated CDIO-SC implementation

CDIO Phase	**Spiral Emphasis**	**Key Learning Activities**	**Primary Outcomes**
Conceive	Initial exposure	Define robotic control problems under uncertainty; introduce fuzzy logic and neural adaptation concepts	Problem formulation; intuition for uncertainty
Design	Conceptual deepening	Design fuzzy rules, membership functions, and neural learning structure; analyze interpretability vs. adaptability	Algorithm modeling; design trade-off analysis
Implement	Reinforcement through practice	Implement and train neuro-fuzzy controller in MATLAB/Python; simulate control performance	Hands-on implementation; data-driven learning
Operate	System integration	Evaluate robustness and stability; refine controller via iterative testing in varied scenarios	System evaluation; engineering judgment

3 The Course Description and Project Design

This paper presents the design and implementation of a CI course in ECE using a hybrid pedagogical approach that integrates the CDIO framework with a SC model. The course is designed to help ECE students develop a balanced mastery of both theoretical foundations and practical applications in CI, including evolutionary computation, swarm intelligence, bio-inspired neural networks, fuzzy systems, and nature-inspired optimization. CI is a multidisciplinary field concerned with theory, design, application, and development of biologically and nature-inspired computational paradigms. Traditionally, CI is founded on three core pillars: neural networks, fuzzy systems, and evolutionary computation. With rapid advances in computing hardware, parallel processing, and memory technologies, CI techniques have become increasingly practical and impactful across a wide range of engineering domains, including robotics, control systems, embedded systems, optimization, and intelligent decision-making.

The CI course described in this paper is offered within the ECE and is intentionally designed to differ from traditional theory-intensive engineering courses. Rather than aiming for exhaustive theoretical coverage of all CI methods, the course selectively emphasizes those techniques most beneficial for engineering applications. Topics include neural networks, bio-inspired systems, fuzzy logic, evolutionary computation, simulated annealing, genetic algorithms, swarm intelligence, and other nature-inspired optimization methods [35]. Given the breadth of CI, the course prioritizes conceptual understanding, modeling, simulation, and practical implementation over mathematical depth in isolated techniques.

Building on the hybrid CDIO-SC pedagogy described in Section 2, the 15-week CI course is implemented through team-based, milestone-driven projects that align module topics with CDIO stages and spiral revisits of core CI concepts. Students define engineering specifications, design algorithmic solutions, implement computational models (e.g., MATLAB/Python), and evaluate performance against explicit criteria, with attention to trade-offs such as robustness, convergence, and computational cost. In this CI course, we assigned a project on robot navigation using the Ant Colony Optimization (ACO) algorithm to teach students how to apply ACO for path planning and routing optimization, as illustrated in Figure 4 using CDIO-SC pedagogical method.

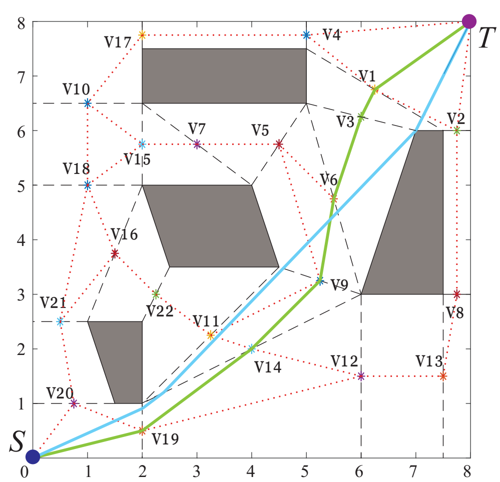

Figure 4: Illustration of simulation results in an environment with multiple obstacles (brown polygons). The graph shows vertices representing waypoints, with connectivity indicated by dashed and dotted lines. The blue line depicts the optimal trajectory generated by the Ant Colony Optimization (ACO) method, demonstrating collision-free navigation from start (S) to target (T) while efficiently avoiding obstacles and minimizing path cost [36–39].

Table 2: Weekly topics, CI techniques, and project-based activities in the CI course

Week	Theme	Core CI Topics	Representative Project	Assessment
1	CI Foundations	CI vs. AI vs. ML; Engineering scope of CI	Case studies of CI applications in ECE	Short reflection
2	CI Applications	CI in robotics, optimization, FPGA/VLSI	Literature replication of CI application	Literature review
3	Search-Based Optimization	Greedy, hill climbing, simulated annealing, tabu search	Implement SA on benchmark problems	Performance comparison
4	Evolutionary Computation I	Genetic Algorithms (GA)	GA-based function optimization	Technical report
5	Evolutionary Computation II	GP, ES, Differential Evolution	Compare GA vs. DE	Algorithm analysis
6	CI in Hardware	CI constraints in FPGA/VLSI	CI hardware feasibility study	Design memo
7	Fuzzy Systems I	Fuzzy logic, sets, membership functions	Design fuzzy inference system	Homework + simulation
8	Fuzzy Systems II	Fuzzy rules, linguistic modeling, defuzzification	Fuzzy control application	Control design report
9	Swarm Intelligence I	Swarm principles, PSO	PSO implementation	Convergence analysis
10	Swarm Intelligence II	ACO, ABC, flocking, foraging	ACO for path planning	Comparative study
11	Bio-inspired NN I	Neuron models, Hebbian learning, SOM	Pattern recognition using NN	Lab report
12	Bio-inspired NN II	Neuro-fuzzy, hybrid CI models	Neuro-fuzzy system design	Case study
13	Advanced CI	SNNs, STDP, adaptive learning	SNN simulation demo	Critical review
14	CDIO Integration	System-level CI integration	Project refinement & testing	Draft project report
15	Capstone Projects	Integrated CI solutions	Project demo & presentation	Final report & presentation

Using the developed framework, various milestone projects and weekly activities (shown in Table 1) are incorporated to cover foundational topics, advanced methods, and ongoing challenges in CI in terms of project/homework assignments. Table 2 summarizes the 15-week structure of the CI course, illustrating how lecture topics, CI techniques, and project-based activities are integrated using a hybrid CDIO-SC framework. The course is intentionally organized to revisit core CI concepts, including optimization, learning, and reasoning under uncertainty, across evolutionary computation, fuzzy systems, swarm intelligence, and bio-inspired neural networks at increasing levels of complexity. Early weeks emphasize conceptual foundations and problem formulation (*Conceive*), followed by algorithmic modeling and design (*Design*), hands-on implementation and experimentation (*Implement*), and system-level evaluation through capstone projects (*Operate*). Representative laboratory activities and projects are aligned with weekly topics to reinforce the connection between theory and engineering practice, requiring students to translate mathematical and computational principles into functional CI systems. Assessments include technical reports, performance analyses, comparative studies, and project presentations, with evaluation criteria mapped to ABET Student Outcomes related to problem solving, system design, experimentation, teamwork, and communication. This structure supports cumulative learning, promotes student engagement, and enables consistent assessment of both technical competencies and professional skills throughout the semester.

4 Assessment Methodologies and Evaluation Results

4.1 Student Learning Outcomes

The explicit alignment between weekly course topics and ABET Student Outcomes (1-7) to demonstrate how the CI course systematically supports accreditation requirements is outlined in Table 3. Problem solving (Outcome 1) and experimentation and data analysis (Outcome 6) are emphasized throughout the semester through algorithm development, simulation, and

Table 3: Mapping of weekly course topics to ABET student outcomes (1–7)

Week	Course Theme	Primary ABET Outcomes Addressed
1	CI Foundations (CI vs. AI vs. ML)	1, 3, 7
2	Engineering Applications of CI	1, 4, 7
3	Search-Based Optimization	1, 6, 7
4	Evolutionary Computation I (GA)	1, 2, 6
5	Evolutionary Computation II (GP, ES, DE)	1, 2, 6, 7
6	CI in FPGA and VLSI Design	1, 2, 4
7	Fuzzy Systems I (Logic and Sets)	1, 6
8	Fuzzy Systems II (Rules and Control)	1, 2, 6
9	Swarm Intelligence I (PSO)	1, 6, 7
10	Swarm Intelligence II (ACO, ABC, WOA)	1, 2, 5, 6
11	Bio-inspired Neural Networks I	1, 6
12	Bio-inspired Neural Networks II (Hybrid CI)	1, 2, 6, 7
13	Advanced CI (SNN, STDP)	1, 7
14	CDIO Project Integration	2, 3, 5, 6
15	Capstone Project & Presentation	2, 3, 4, 5

performance evaluation across multiple CI paradigms. Engineering design (Outcome 2) is progressively strengthened as students move from individual algorithm implementation to integrated, system-level CI solutions in the capstone project. Communication (Outcome 3) and teamwork (Outcome 5) are reinforced through team-based projects, technical reports, and formal presentations, particularly during the CDIO integration and capstone phases. Ethical and professional considerations (Outcome 4) are addressed in application-driven discussions involving hardware constraints, societal impact, and responsible AI use. Lifelong learning and the ability to acquire new knowledge (Outcome 7) are fostered through exposure to emerging CI techniques and independent exploration of current research. This structured mapping provides clear evidence of continuous and intentional ABET outcome coverage throughout the course.

4.2 Student Survey Design and Cross-Semester Comparison

To clarify the design of the three student surveys administered during the semester, it is important to describe the assessment rationale. Under the hybrid CDIO-SC instructional approach, student development was expected to occur progressively rather than abruptly. The pedagogy emphasizes iterative reinforcement and increasing conceptual depth; accordingly, incremental growth across the semester aligns with the intended learning trajectory. The course was not divided into experimental and control groups within the same semester; all students experienced the new instructional approach. This design choice was intentional to prevent inequities in learning opportunities and avoid negatively affecting students who might not receive the redesigned pedagogy. From an ethical and educational standpoint, the new approach was applied uniformly to the entire class. Comparative analysis was conducted across semesters (see 4). Specifically, student performance and survey responses from the semester implementing the CDIO-SC hybrid pedagogy were compared with prior offerings of the same course, which followed a more traditional structure without systematic CDIO milestone alignment or spiral reinforcement. This cross-semester comparison provides additional context for evaluating the potential impact of the

Table 4: Cross-semester comparison of survey means (project-based learning vs. CDIO-SC)

Project-based learning	**Questionnaire 1**	**Questionnaire 2**	**Questionnaire 3**
Mean score	3.86	3.97	3.73
Improvement based on Questionnaire 1	—	2.85%	-3.37%
Improvement based on Questionnaire 2	—	—	-6.05%
CDIO-SC	**Questionnaire 1**	**Questionnaire 2**	**Questionnaire 3**
Mean score	4.01	4.17	4.36
Improvement based on Questionnaire 1	—	3.99%	8.73%
Improvement based on Questionnaire 2	—	—	4.56%

redesigned instructional approach. Within the semester implementing the new pedagogy, survey results collected at three stages of the course indicate a steady, incremental upward trend in student self-assessed competency. The improvement is gradual rather than abrupt, reflecting cumulative learning over time. This pattern is consistent with the core principle of the spiral curriculum, in which knowledge and skills are revisited and deepened progressively across instructional stages.

While the within-semester survey results illustrate structured learning progression, cross-semester comparisons provide a broader reference for assessing instructional impact. Together, these findings suggest that the hybrid CDIO-SC design supports sustained development of computational intelligence competencies rather than short-term gains. Future work will strengthen this analysis through expanded cross-cohort data and additional objective performance metrics.

4.3 Stage-Level Performance Trends

In Table 5, we present the aggregated student performance across the four CDIO phases. The overall score for each phase was calculated by compiling all assessment components associated with that phase, summing the final scores of all students, and computing the average value. These four phase-level averages were then compared to examine performance trends throughout the course. The overall trend indicates a steady improvement in student performance across the four stages, which is consistent with the intended scaffolding of the CDIO-SC design. However, it is noteworthy that the magnitude of improvement in the fourth phase is smaller than in the earlier phases. One primary reason is that the first three phases did not include presentation evaluation components, whereas the fourth phase incorporated both the final examination and the final project presentation scores. These additional components lowered the overall average, particularly because oral presentations represent a relative weakness for many students. Overall, the results suggest a consistent upward trajectory in student performance. From one perspective, this trend provides supporting evidence that the adoption of the new pedagogical approach has contributed to improved student learning outcomes.
The aggregated student performance across the four instructional stages aligned with the CDIO framework and the spiral curriculum structure is presented in Table 5. The reported stage-level averages (75%, 81%, 87%, and 91%) demonstrate a consistent upward trajectory as students progressed through increasingly advanced computational intelligence topics. Several observations can be drawn from this trend. First, the steady increase in average performance across stages is

Table 5: Student learning assessment based on homeworkand milestone project performance

Stage	Materials Covered	Measurement	Assessment	Action for Improvement
Stage 1	CI vs. AI vs. ML CI for engineering applications CI in robotics, optimization, FPGA/VLSI Greedy, simulated annealing, tabu search	HW assignments In-class exercise Interview Milestone project 1	**75%**	Debugging skills Analytical skills Problem-solving skills Modeling & simulation skills
Stage 2	Genetic Algorithms (GA) GP, ES, Differential Evolution GA for engineering applications	HW assignments In-class exercise; Interview; Test 1 Milestone project 2 Student presentation	**81%**	Debugging skills Technical writing skills Problem-solving skills Analytical skills
Stage 3	Swarm intelligence, PSO ACO, ABC, flocking, foraging Firefly Algorithm Whale Optimization Algorithm Dragonfly Algorithm	HW assignments In-class exercise Interview Test 2 Milestone project 3	**87%**	Problem-solving skills Collaboration skills Debugging skills
Stage 4	Fuzzy set theory Membership functions Fuzzy inference systems Type-2 fuzzy systems ANFIS layer structure Neuro-fuzzy systems	HW assignments In-class exercise Interview Final exam Final project Final presentation	**91%**	Student presentation Problem-solving skills Technical writing skills Collaboration skills Modeling & simulation skills Failure analysis

consistent with the intended cumulative learning model of the spiral curriculum. In this instructional design, foundational search algorithms introduced in Stage 1 (e.g., greedy search, hill climbing, simulated annealing, tabu search) are conceptually revisited and extended in subsequent stages through evolutionary computation (Stage 2), swarm intelligence (Stage 3), and fuzzy/neuro-fuzzy systems (Stage 4). Each stage builds upon prior algorithmic principles such as exploration versus exploitation, convergence behavior, stochastic modeling, and parameter tuning. The observed progression in performance suggests that iterative reinforcement and conceptual revisiting may support deeper comprehension and improved technical execution. Second, the CDIO-based milestone structure likely contributes to progressive skill development. Each stage incorporates homework, in-class exercises, interviews, and milestone projects aligned with Conceive-Design-Implement-Operate cycles.

As students move through the stages, the cognitive demands increase from algorithm implementation toward system-level integration and hybrid intelligent system design. The continued improvement in performance despite increasing task complexity suggests growth not only in content mastery but also in higher-order competencies such as modeling, debugging, collaboration, and system integration. Third, Stage 4 includes broader and more comprehensive assessment components, including the final examination and final project presentation. These evaluation elements require synthesis of multiple computational intelligence paradigms and integration into functional systems. The fact that the stage-level average remains the highest (91%) even under expanded assessment scope indicates that students were able to adapt to increased rigor and complexity. This outcome aligns with the course's objective of developing both analytical depth and implementation maturity. It is important to clarify that the stage-level progression reflects cumulative performance within a structured semester sequence rather than a controlled experimental comparison. Therefore, while the upward trend is consistent with the instructional scaffolding embedded in the hybrid CDIO-SC design, the results should be interpreted as evidence of structured learning progression rather than definitive causal proof of pedagogical superiority. Future iterations of the course will incorporate cross-cohort comparisons and additional objective metrics to further strengthen empirical validation. Overall, the results in Table 5 suggest that aligning computational intelligence topics with spiral reinforcement and CDIO milestone implementation may support sustained performance growth across increasingly complex algorithmic domains. The trend reflects not only content acquisition but also the development of professional engineering competencies aligned with ABET student outcomes, including problem-solving, design, communication, and teamwork.

4.4 Assessment and Project Evaluation

To rigorously examine the educational impact of the CDIO-SC hybrid pedagogy framework, a multi-dimensional assessment design is employed in the Computational Intelligence course. This design integrates formative and summative assessments, structured self-assessments, and ongoing feedback loops to systematically evaluate both conceptual learning and applied competence. The assessment strategy is informed by established engineering education research practices and emphasizes triangulation across multiple data sources.

Pre- and Post-Semester Self-Assessments: Students complete structured self-assessment instruments at the beginning and conclusion of the semester. These instruments are explicitly aligned with ABET student learning outcomes and are designed to measure targeted constructs, including engineering problem solving, algorithmic reasoning, technical implementation skills, and the ability to transfer engineering principles to complex, open-ended robotic navigation problems. Paired pre- and post-semester analyses enable the examination of learning gains and shifts in student self-efficacy, providing quantitative evidence of cognitive and affective development associated with engagement in sampling-based robot path planning activities.

Milestone-Based Project Evaluation: Grounded in formative assessment theory, the CDIO-SC hybrid pedagogy framework employs structured, milestone-based project evaluations in which students submit incremental project deliverables at multiple points throughout the semester. Each milestone functions as a formative checkpoint and is evaluated using analytically designed rubrics that provide actionable feedback on technical correctness, algorithmic implementation, and design decisions. Beyond technical performance, the rubrics assess students' adaptability, creative problem solving, and responsiveness to feedback, emphasizing learning as an iterative process. By embedding timely feedback and opportunities for revision, this approach supports continuous learning improvement and enables systematic tracking of student development in complex, open-ended robotics problem solving tasks using CI algorithms.

Reflective Q&A and Peer Interviews: Structured reflective Q&A sessions and end-of-project peer interviews are incorporated as qualitative assessment components to elicit students' perspectives on their learning experiences. These reflective activities provide rich qualitative data on students perceived learning gains, conceptual challenges, and engagement with the instructional design. The resulting insights are systematically analyzed to inform instructional refinement and to guide in iterative improvements in subsequent offerings of the course.

By integrating adaptive feedback mechanisms with structured evaluation instruments, the proposed assessment strategy supports the personalized and iterative learning processes central to the CDIO-SC framework. This design not only facilitates continuous instructional refinement but also produces actionable evidence to inform educational research and course improvement. To enhance students' understanding of CI algorithms for robot navigation, a series of carefully designed review assignments was embedded throughout the CI course. These assignments were integrated into milestone-based project reviews and functioned as structured opportunities for self-assessment and continuous learning within the project-based instructional framework.

Three self-assessment questionnaires were administered over the semester, with results summarized in Table 4. Using a five-point Likert scale, the average student score increased from 4.01 in Questionnaire 1 to 4.17 in Questionnaire 2, representing a 3.99% improvement relative to the baseline. Questionnaire 3 further increased to an average score of 4.36, corresponding to an 8.73% improvement over Questionnaire 1 and a 4.56% gain relative to Questionnaire 2. This consistent upward trend indicates progressive enhancement in students' perceived understanding and confidence by using this CDIO-SC hybrid pedagogy. Collectively, these results provide empirical support for the effectiveness of the CDIO-SC framework in promoting sustained learning gains through iterative feedback and scaffolded instructional design.

Preliminary results show that students demonstrate enhanced competencies in CI algorithm design, system integration, and problem-solving. They are increasingly capable of translating mathematical and computational principles into engineering solutions for robotics, control, and embedded systems applications. Evidence from assignments, Q&A sessions, examinations, ABET-style assessments, self-assessment surveys, and formal course evaluations compared to results from previous years indicates that the hybrid CDIO-SC pedagogy significantly improves engagement, conceptual retention, and hands-on learning outcomes. This work demonstrates that integrating CDIO with a SC provides an effective and scalable instructional model for CI education. The approach develops both technical expertise and professional competencies, including critical thinking, innovation, teamwork, and system-level reasoning skills essential for modern engineers who integrate AI and CI to solve interdisciplinary challenges. Future work will expand longitudinal ABET assessments, foster interdisciplinary collaboration, and disseminate modular CI project materials for broader adoption across engineering curricula.

In this course, students respond to five self-assessment questions, each explicitly aligned with designated ABET student outcomes. The following questions focus on competencies related to sampling-based robot motion planning:

- Question 1: "*I can understand most questions/problems assigned with Milestone Project on Genetic Algorithms, and Differential Evaluation and I can design, analyze and complete the simulated experiments in this milestone project , based on required prerequisites by topics such as engineering programming (MATLAB, C/C++, or Python), calculus and differential equations, probability, linear algebra/matrix algebra listed in syllabus.*"
  Aligned with ABET Outcome (b): An ability to design and conduct experiments, as well as to analyze and interpret data related to electrical systems.

- Question 2: "*I am able to design and implement nature-inspired algorithms such as ACO and PSO algorithms to generate collision-free paths that satisfy specific task requirements and environmental constraints.*"
  Aligned with ABET Outcome (c): An ability to design electrical systems, components, or processes to meet desired needs.

- Question 3: "*I am able to identify and clearly formulate bio-inspired optimalization problems and apply appropriate bio-inspired optimalization algorithms to develop effective solutions (such as robot navigation, FPGA, etc.), including implementing and validating these solutions using MATLAB.*"
  Aligned with ABET Outcome (e): An ability to identify, formulate, and solve electrical engineering problems.
- Question 4: "*I demonstrate effective communication skills when working collaboratively in a multidisciplinary team environment involving students from Computer Engineering, Electrical Engineering, Software Engineering, Mechanical Engineering, Agricultural and Biological Engineering, and Computer Science.*"
  Aligned with ABET Outcome (g): An ability to communicate effectively.

Table 6: The questionnaire of students for assessment of education quality

Questions and Outcome	Survey				
	Strongly agree	Agree	Neutral	Disagree	Strongly disagree
Q1 – (*b*)	39.5%	53.5%	7.0%	0%	0%
Q2 – (*c*)	46.5%	32.6%	16.3%	4.6%	0%
Q3 – (*e*)	37.2%	41.9%	10.8%	10.1%	0%
Q4 – (*g*)	56.3%	22.5%	9.3%	5.4%	6.5%

The self-assessment questionnaire results are summarized in Table 6. Overall, students responded positively, with the majority selecting "strongly agree" or "agree" across all questions aligned with the designated ABET outcomes. For Question 1 (Outcome b), which assesses the ability to design and conduct experiments and analyze electrical system data, 39.5% of *students strongly* agreed and 53.5% *agreed*, resulting in a combined 93.0% agreement. This high percentage indicates that students felt confident in applying prerequisite knowledge to milestone projects, suggesting that the hybrid, student-centered pedagogical approach effectively supported experimental design and analysis skills. Another contributing factor is that the instructor dedicated time to teaching Genetic Algorithms and Differential Evolution algorithms using the CDIO-SC framework and milestone-driven pedagogy, emphasizing both technical proficiency and professional competencies.

For Question 2 (Outcome c), which assesses students' ability to design and implement nature-inspired algorithms, 46.5% *strongly agreed* and 32.6% *agreed*. A small fraction (4.6%) disagreed, which may reflect limited hands-on opportunities in some simulation exercises, particularly as students faced challenges with MATLAB troubleshooting and had varying levels of programming experience.

Question 3 (Outcome e), assessing problem identification, formulation, and solution in bio-inspired optimization contexts, received 37.2% *strongly agree* and 41.9% *agree* responses, indicating substantial student confidence in applying algorithmic solutions to complex engineering problems. Bio-inspired optimization algorithms were emphasized extensively using the CDIO-SC framework. The instructor demonstrated these algorithms in class, providing clear examples of their implementation and application. In addition, homework assignments required students to work with heuristic optimization models, such as Simulated Annealing (SA) and Hill Climbing (HC), allowing them to practice and deepen their understanding of optimization techniques. As a result, students were able to engage hands-on with both bio-inspired and classical heuristic optimization methods, reinforcing conceptual understanding and practical proficiency in algorithm design.

Interestingly, Question 4 (Outcome g), which assessed communication skills in a multidisciplinary team environment, showed 56.3% *strongly agree* and 22.5% *agree*, indicating that most students perceived strong teamwork and collaborative communication skills. Neutral responses accounted for 9.3%, while 5.4% disagreed and 6.5% strongly disagreed. The disagreement reflects the diversity of student majors in the class; during discussions, some students were unfamiliar with peers from other disciplines, which may have impacted their perception of communication effectiveness. Compared to previous iterations of the course taught using traditional lecture-based methods, the combined percentages of "strongly agree" and "agree" are higher across all ABET outcomes, demonstrating that the CDIO-SC framework and milestone-driven pedagogy positively influence both technical and professional competencies, including collaboration and interdisciplinary communication.

Therefore, the results suggest that integrating computational intelligence topics with spiral reinforcement and CDIO milestone implementation supports sustained performance growth across increasingly complex algorithmic domains. The trend reflects both conceptual acquisition and professional skill development aligned with ABET student outcomes, including problem-solving, design, communication, and teamwork.

5 Conclusion

This paper presented the design and implementation of a CI course using a hybrid pedagogical approach that integrates the CDIO framework with a SC model. By combining project-driven learning with iterative conceptual reinforcement, the CDIO-SC approach effectively bridges theoretical foundations and practical applications across core CI topics, including evolutionary computation, swarm intelligence, fuzzy systems, and bio-inspired neural networks. Course activities and milestone-based projects guided students through the full engineering lifecycle while progressively deepening their conceptual understanding. Assessment results from assignments, self-assessments, ABET-aligned evaluations, and course feedback indicate improved student engagement, conceptual retention, and applied problem-solving skills compared to prior course offerings. The findings suggest that the CDIO-SC framework provides a scalable and adaptable instructional model for CI education. This approach supports the development of both technical proficiency and professional competencies essential for preparing undergraduate engineers to apply AI-driven methods in complex, real-world engineering systems.

References

[1] L. Wang, C. Luo, M. Li, and J. Cai, "Trajectory planning of an autonomous mobile robot by evolving ant colony system," *International Journal of Robotics and Automation*, vol. 32, no. 4, pp. 1500–1515, 2017.

[2] M. Hicks, T. Lei, C. Luo, L. Liu, and Z. Bi, "A bio-inspired goal-directed cognitive map approach to robot navigation and mapping," in *2025 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, 2025, pp. 1–8.

[3] T. Sellers, T. Lei, C. Luo, Z. Bi, and G. E. Jan, "A dynamically updating graph-based navigation scheme for autonomous vehicles," *Sensors and AI*, vol. 2, no. 1, pp. 1–16, 2026.

[4] T. Lei, T. Sellers, C. Luo, Z. Bi, and G. E. Jan, "Developing computational intelligence curriculum materials to advance student learning for robot control and optimization," in *ASEE Annual Conference & Exposition*, 2024.

[5] P. Chintam, T. Lei, B. Osmanoglu, Y. Wang, and C. Luo, “Informed sampling space driven robot informative path planning,” *Robotics and Autonomous Systems*, vol. 175, p. 104656, 2024.

[6] D. Short, T. Lei, D. W. Carruth, C. Luo, and Z. Bi, “A bio-inspired algorithm in image-based path planning and localization using visual features and maps,” *Intelligence & Robotics*, vol. 3, no. 2, pp. 222–41, 2023.

[7] M. Hicks, T. Lei, C. Luo, D. W. Carruth, and Z. Bi, “A bio-inspired goal-directed cognitive map model for robot navigation and exploration,” *IEEE Transactions on Cognitive and Developmental Systems*, vol. 17, no. 5, pp. 1125–1140, 2025.

[8] T. Lei, P. Chintam, C. Luo, L. Liu, and G. E. Jan, “A convex optimization approach to multi-robot task allocation and path planning,” *Sensors*, vol. 23, no. 11, p. 5103, 2023.

[9] T. Sellers, T. Lei, C. Luo, Z. Bi, and G. E. Jan, “Human autonomy teaming-based safety-aware navigation through bio-inspired and graph-based algorithms,” *Biomimetic Intelligence and Robotics*, vol. 4, no. 4, p. 100189, 2024.

[10] G. E. Jan, T. Lei, C.-C. Sun, Z.-Y. You, and C. Luo, “On the problems of drone formation and light shows,” *IEEE transactions on consumer electronics*, vol. 70, no. 3, pp. 5259–5268, 2024.

[11] J. London, T. Lei, Y. Yuan, S. Shao, and J. Hu, “Enhanced biomedical signal processing using time-frequency analysis and wavelet transforms,” in *2025 IEEE 68th International Midwest Symposium on Circuits and Systems (MWSCAS)*. IEEE, 2025, pp. 463–467.

[12] T. Lei, T. Sellers, C. Luo, L. Gong, Z. Bi, and J. Wang, “ToP-CK: Technological pedagogical content knowledge and adaptive feedback integrated robot path planning,” in *2025 IEEE Frontiers in Education Conference (FIE)*. IEEE, 2025, pp. 1–9.

[13] M. Menekse, G. S. Stump, S. Krause, and M. T. Chi, “Differentiated overt learning activities for effective instruction in engineering classrooms,” *Journal of Engineering Education*, vol. 102, no. 3, pp. 346–374, 2013.

[14] J. London, T. Lei, C. Luo, and Y. Yuan, “Human activity recognition and biomedical signal classification using multi-modal deep learning,” in *2025 IEEE 68th International Midwest Symposium on Circuits and Systems (MWSCAS)*. IEEE, 2025, pp. 676–680.

[15] T. Lei, T. Sellers, U. Iqbal, J. Wang, and C. Luo, “A hybrid learning approach to multi-target navigation in robotics education,” in *2025 IEEE Frontiers in Education Conference (FIE)*. IEEE, 2025, pp. 1–9.

[16] T. Sellers, T. Lei, J. Wang, U. Iqbal, L. Liu, and C. Luo, “Advancing robot local navigation through vector field histograms using a hybrid learning approach,” in *2025 IEEE Frontiers in Education Conference (FIE)*. IEEE, 2025, pp. 1–9.

[17] B. Kollöffel and T. De Jong, “Conceptual understanding of electrical circuits in secondary vocational engineering education: Combining traditional instruction with inquiry learning in a virtual lab,” *Journal of Engineering Education*, vol. 102, no. 3, pp. 375–393, 2013.

[18] W. Zhao, X. Luo, C. Luo, and Y. Peng, “Design and implementation of project-based courses on cutting-edge computer technologies,” in *2017 ASEE Annual Conference & Exposition*, 2017.

[19] N. Hosseinzadeh and M. R. Hesamzadeh, “Application of project-based learning (PBL) to the teaching of electrical power systems engineering,” *IEEE Transactions on Education*, vol. 55, no. 4, pp. 495–501, 2012.

[20] J. Rodríguez, A. Laveron-Simavilla, J. M. del Cura, J. M. Ezquerro, V. Lapuerta, and M. Cordero-Gracia, “Project based learning experiences in the space engineering education at Technical University of Madrid,” *Advances in Space Research*, vol. 56, no. 7, pp. 1319–1330, 2015.

[21] L. Cheng, A. D. Ritzhaupt, and P. Antonenko, “Effects of the flipped classroom instructional strategy on students’ learning outcomes: A meta-analysis,” *Educational Technology Research and Development*, vol. 67, pp. 793–824, 2019.

[22] E. A. Van Vliet, J. C. Winnips, and N. Brouwer, “Flipped-class pedagogy enhances student metacognition and collaborative-learning strategies in higher education but effect does not persist,” *CBE—Life Sciences Education*, vol. 14, no. 3, p. ar26, 2015.

[23] T. Sellers, T. Lei, C. Luo, G. E. Jan, and Z. Bi, “A hybrid pedagogy through topical guide objective to enhance student learning in MIPS instruction set design,” in *ASEE Annual Conference & Exposition*, 2024.

[24] E. Marasco and L. Behjat, “Integrating creativity into elementary electrical engineering education using CDIO and project-based learning,” in *2013 IEEE International Conference on Microelectronic Systems Education (MSE)*, 2013, pp. 44–47.

[25] S. J. Dickerson and R. M. Clark, “Implementation of lessons learned to simulation-based reflection in a digital circuits course.” *Computers in Education Journal*, vol. 13, 2022.

[26] J. Wang, C. Luo, W. Zhao, and X. Li, “Empowering students with self-regulation in a project-based embedded systems course,” in *2017 ASEE Annual Conference & Exposition*, 2017.

[27] T. Lei, T. Sellers, C. Luo, G. E. Jan, and Z. Bi, “An enhanced learning method used for datapath design topics in computer engineering curriculum,” in *ASEE Annual Conference & Exposition*, 2024.

[28] T. Sellers, T. Lei, C. Luo, Z. Bi, and G. E. Jan, “Enhancing student learning in robot path planning optimization through graph-based methods,” in *ASEE Annual Conference & Exposition*, 2024.

[29] T. Lei, T. Sellers, C. Luo, L. Cao, and Z. Bi, “Digital twin-based multi-objective autonomous vehicle navigation approach as applied in infrastructure construction,” *IET Cyber-Systems and Robotics*, vol. 6, no. 2, p. e12110, 2024.

[30] T. Sellers, T. Lei, C. Luo, G. E. Jan, and J. Ma, “A node selection algorithm to graph-based multi-waypoint optimization navigation and mapping,” *Intelligence & Robotics*, vol. 2, no. 4, pp. 333–354, 2022.

[31] L. Dela Cruz, T. Lei, C. Luo, Z. Bi, and G. E. Jan, “Advanced multi-target navigation using add-rrvs with double q-learning,” in *2025 International Conference on Information and Automation (ICIA)*, 2025, pp. 448–453.

[32] T. Sellers, T. Lei, G. E. Jan, Y. Wang, and C. Luo, “Multi-objective optimization robot navigation through a graph-driven PSO mechanism,” in *Advances in Swarm Intelligence: 13th International Conference, ICSI 2022, Xi'an, China, July 15–19, 2022, Proceedings, Part II*. Springer, 2022, pp. 66–77.

[33] T. Sellers, T. Lei, C. Luo, E. Yang, and Z. Bi, “Human-autonomy teaming in robot navigation using bioinspired approach for search and rescue,” *IEEE Transactions on Human-Machine Systems*, 2026.

[34] C. Luo, Y.-T. Wu, M. Krishnan, M. Paulik, G. E. Jan, and J. Gao, “An effective search and navigation model to an auto-recharging station of driverless vehicles,” in *2014 IEEE Symposium on Computational Intelligence in Vehicles and Transportation Systems (CIVTS)*. IEEE, 2014, pp. 100–107.

[35] S. Alarabi, T. Lei, M. Santora, C. Luo, and T. Sellers, “Multi-robot path planning using potential field-based simulated annealing approach,” in *SPIE Conference: Unmanned Systems Technology XXVI*, vol. 13055. SPIE, 2024, pp. 102–117.

[36] T. Lei, T. Sellers, C. Luo, D. W. Carruth, and Z. Bi, “Graph-based robot optimal path planning with bio-inspired algorithms,” *Biomimetic Intelligence and Robotics*, p. 100119, 2023.

[37] S. Steen, T. Lei, C. Luo, S. Shao, and L. Gong, “A moss growth optimization approach to robot path planning,” in *2025 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, 2025, pp. 1–4.

[38] T. Lei, C. Luo, J. E. Ball, and S. Rahimi, “A graph-based ant-like approach to optimal path planning,” in *2020 IEEE congress on evolutionary computation (CEC)*. IEEE, 2020, pp. 1–6.

[39] J. H. Rogers III, T. Sellers, T. Lei, C. R. Hudson, and C. Luo, “Centroid-based cell decomposition robot path planning algorithm integrated with a bio-inspired approach,” in *SPIE Conference: Unmanned Systems Technology XXVI*, vol. 13055. SPIE, 2024, pp. 43–51.