Sensors and Materials, Vol. 36, No. 3 (2024) 1147–1161
MYU Tokyo

S & M 3588

ChatGPT-powered Inquiry-based Learning Model of Training for Intelligent Car Racing Competition

Qiang Chen,[1] Hung-Cheng Chen,[2*] and Yu-Liang Lin[3]

[1]School of Railway, Shandong Polytechnic,
No. 23000, Jingshidong Road, Jinan City, Shandong Province 250104, China
[2]School of Education, Huanggang Normal University,
No. 146, Xingang Second Road, Development Zone, Huanggang City, Hubei Province 438000, China
[3]School of Urban Rail, Shandong Polytechnic,
No. 23000, Jingshidong Road, Jinan City, Shandong Province 250104, China

(Received October 20, 2023; accepted March 18, 2024)

Keywords: inquiry-based learning, ChatGPT, intelligent car racing competition, project-based learning, prompt engineering

In this study, we explore the application of an inquiry-based learning model powered by ChatGPT in the context of intelligent car racing competition training. We address four key aspects: (1) the construction of a knowledge and skill acquisition process through student interactions with ChatGPT to facilitate the progressive development of problem-solving strategies and approaches; (2) project-based learning for interdisciplinary students participating in the competition, where students are grouped in accordance with their backgrounds and engage in tasks such as vehicle design and optimization, electrical drive and control algorithm adaptation, and sensor circuit design and calibration; (3) the paradigm shift in the role of teachers, transitioning from knowledge providers to co-coaches alongside ChatGPT, allowing teachers to allocate more time to monitor the progress of different student groups and design learning objectives; and (4) knowledge building and prompt engineering during different stages of the training process, where students employ various questions and prompts to interact with ChatGPT, thereby constructing domain-specific knowledge and improving the quality and effectiveness of knowledge acquisition. By leveraging ChatGPT as a conversational agent, students engage in a dynamic learning process that fosters their understanding of research problems and nurtures their problem-solving skills. Integrating an inquiry-based approach, project-based learning, and teacher-student collaboration with ChatGPT empowers students to acquire essential knowledge and cultivate critical thinking abilities, contributing to their overall growth and readiness for intelligent car racing competitions. The findings of this study shed light on the efficacy of ChatGPT-powered inquiry-based learning models in preparing students for complex and interdisciplinary challenges in the field of intelligent car racing.

*Corresponding author: e-mail: hungcheng@hgnu.edu.cn
https://doi.org/10.18494/SAM4726

ISSN 0914-4935

1. Introduction

1.1 Brief introduction of National Intelligent Car Racing Competition for College Students in China

The National Intelligent Car Racing Competition for College Students, which started in 2006, is a scientific and technological competition organized by the Ministry of Education of China, focusing on the research of intelligent vehicles (Fig. 1). This competition attracts hundreds of universities to participate yearly. It has been held successfully 17 times with more than 420,000 participants. The competition is divided into indoor and outdoor races run on several tracks. Indoor tracks include a photoelectric group, a camera group, an electromagnetic group, an electric rail group, and others. Outdoor tracks include groups such as LIDAR, wireless, and driverless groups. The aim of the competition is to cultivate students' comprehensive knowledge competency, engineering practice ability, and innovative spirit. This event has become an essential platform for Chinese university students to study and research automation, which not only provides an opportunity to exhibit their creative achievements but also gives students a chance to communicate, learn from each other, and understand the development of the industry.[1]

1.2 Inquiry-based learning

Inquiry-based learning is a student-centered approach to teaching and learning that allows students to take the initiative to ask questions, practice inquiry, and gain knowledge and competence.[2–5] It can be divided into five stages:[2] (1) orientation - guiding students to develop a basic understanding of the inquiry topic; (2) conceptualization - students formulate questions or hypotheses; (3) investigation - students gather data and information to answer questions or

Fig. 1. (Color online) National Intelligent Car Racing Competition 2023 at Tiangong University.

test hypotheses; (4) conclusion - students analyze data and draw conclusions; (5) discussion - students exchange and share results and experiences. These five stages are iterative processes through which students deepen their understanding through repeated cycles. Inquiry-based learning fosters self-directed learning but requires substantial teacher guidance and resource support. It emphasizes students' active participation and problem-driven approaches to acquiring knowledge. Through the facilitation of cooperation, communication, and practical application, inquiry-based learning cultivates the students' spirit of inquiry, innovation, and lifelong learning.

1.3 ChatGPT for education

ChatGPT, an AI chatbot developed by OpenAI, has gained popularity since its public release in 2022. Considerable discussion has focused on the potential applications of ChatGPT in educational settings and the challenges in practical implementation.(6) Significant attention has been paid to how AI, such as ChatGPT, can be successfully integrated into classrooms to benefit teachers and students, including responsible and ethical use. The following are some highlighted educational applications of ChatGPT: (1) Enhancing learning effectiveness: ChatGPT can help students express ideas clearly and coherently, allowing instructors to focus on the ideas themselves during teaching and discussion. (2) Serving as an effective teaching aid: ChatGPT can summarize arguments, knowledge, and concepts to support formative assessment during instruction. (3) Improving programming skills: In computer science courses, instructors can use ChatGPT to interpret an existing code, generate a new code, and significantly enhance the students' learning efficiency and sense of achievement.(7) More research is still needed to fully realize the potential of AI in education while addressing concerns about proper oversight and limitations.(8–11)

In this study, we showcase a novel learning model that uses ChatGPT to enhance the training process in intelligent car racing competitions. The approach is aimed at developing problem-solving strategies through iterative student interactions with ChatGPT, integrating project-based learning and teacher-student collaboration. The innovative approach highlights AI's potential in education to prepare students for complex, interdisciplinary challenges in the intelligent car racing domain. The significant aspect of this study is the practical application of ChatGPT in fostering a dynamic and collaborative learning environment.

2. Inquiry-based Learning in Intelligent Car Racing Competition Training

2.1 Design principles of intelligent vehicles

As described in this paper, an intelligent vehicle is used in a competition, for which participants must determine optimal control strategies. The vehicle primarily comprises sensors, motor drives, steering servos, and programmed controllers (Fig. 2). Competitors leverage the characteristics of electromagnetic signals generated from vehicle–road interactions to determine road conditions and decide control approaches.

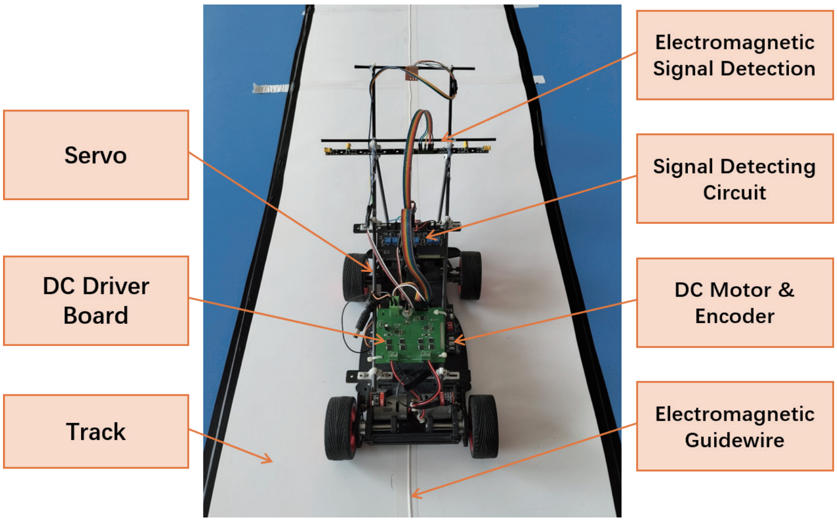


Fig. 2. (Color online) Schematic of electromagnetically driven intelligent vehicle.

The intelligent vehicle used in the competition employs electromagnetic navigation principles to follow the track. The workflow entails the following steps: (1) installing electromagnetic coils on the track surface to form an electromagnetic field; (2) outfitting the vehicle with electromagnetic sensors to detect signals in real time; (3) calculating the vehicle's lateral offset relative to the track per signal; (4) computing steering rudder angles via a proportional-integral-derivative (PID) controller to recenter the vehicle; and (5) achieving precise navigation through continuous detection and feedback control.

The electromagnetic sensor is an electromagnetic induction sensor consisting of I-shaped inductors (model: TE Connectivity SC1314) in parallel with capacitors (model: Murata 6.8nF ceramic capacitors). This sensor can detect the strength and direction of magnetic fields, thereby determining the position and orientation of the electromagnetic guidewire. In intelligent vehicle competitions, multiple electromagnetic sensors are typically installed under the vehicle to enhance magnetic signals. Sensitivity is adjusted in accordance with the track, with signals processed to guide direction and speed. During the race, sensors continuously adapt to the changing conditions.

The electromagnetic sensor primarily consists of conductive copper wire coils, insulation materials such as polyester film, core materials such as silicon steel or iron–nickel alloys, and casings and structural components made of plastics, ceramics, or metals, ensuring the mechanical strength and protection of internal components.

The electromagnetic guidewire is an enameled wire of 0.1–1.0 mm diameter laid along the centerline, conducting a 20 kHz, 100 mA alternating current. The frequency range is 20 ± 1 kHz and the current range is 100 ± 20 mA.

The PID controller integrates proportional, integral, and differential parameters to govern vehicle speed, position, and direction. The proportional term controls the speed, the integral controls the position, and the differential handles the direction. Additionally, various sensors beyond cameras are permitted for detection.[12–15]

In summary, the intelligent vehicle in this study achieves precise track following by sensing electromagnetic signals and combining electromagnetic navigation and closed-loop PID control. Participants must optimize the vehicle sensing capabilities and power to achieve the best tracking. This approach to electromagnetic driving provides an accurate and reliable technical platform for intelligent vehicle competitions. Critically optimizing sensor layouts and PID control parameters proves to be the pivotal task.

2.2 Inquiry-based learning with ChatGPT in intelligent vehicle training

Inquiry-based learning using ChatGPT can be implemented in intelligent vehicle competition training. First, teachers design inquiry tasks such as researching sensor technologies or PID control algorithms to stimulate student interest. Teachers provide advice to support students in progressively mastering relevant knowledge and skills while encouraging independent thinking. Second, work groups are organized to facilitate communication and collaborative learning. Independent learning is promoted to develop innovative thinking and problem-solving abilities.

Third, students are guided to use ChatGPT for inquiry-based learning and self-directed knowledge acquisition. Building upon the five stages of inquiry-based learning proposed in Ref. 2, in this study, we elaborate more on inquiry-based learning.

(1) Orientation: Teachers provide scaffolding as an orientation to help students use ChatGPT effectively and avoid misinformation.
(2) Conceptualization: Cultivate critical thinking by prompting students to question and analyze ChatGPT responses.
(3) Investigation: Information literacy skills are honed so that students can identify potential errors or biases.
(4) Conclusion: Collaborative discussions aid student learning and conclusion-making.
(5) Discussion: Iterative reflection and adjustment optimize the learning model and implementation.

In summary, this approach synergizes ChatGPT usage and student-driven inquiry. As described in Ref. 9, teacher scaffolding, critical thinking, and emphasis on communication foster knowledge acquisition and problem-solving skills.

2.3 Interdisciplinary collaboration in group learning

We implement group learning, with each group comprising 3 to 4 students. The students' majors include vehicle, electronics, information, and mechanical and electrical engineering. Instructors assign suitable tasks based on each student's expertise. For instance, vehicle engineering and mechanical engineering students analyze vehicle structure and power traits, while electronic, information, mechanical, and electrical engineering students focus on circuit design and programming. Teachers provide knowledge guidance, resources, and practical instruction. Students are encouraged to inquire independently per their expertise to gain knowledge and solve problems. Our proposed model capitalizes on diverse specializations, enabling knowledge sharing and integration. It cultivates teamwork, hands-on skills, and real-world problem-solving skills while increasing engagement. Interdisciplinary collaboration provides a comprehensive understanding of intelligent vehicles and simulates actual engineering project cooperation.

3. Inquiry-based Learning for PID Controller Optimization

3.1 Building knowledge of PID controller by ChatGPT

ChatGPT guides electronic, information, and electromechanical engineering students to implement PID control algorithms for precise and intelligent vehicle regulation. The PID algorithm synergistically combines positional and incremental (speed) styles to govern servos and motors, ensuring smooth, intelligent vehicle operation.

Specifically, positional PID tunes the servo angles by setting a target travel angle, sensing the real-time deviation between the vehicle body and the electromagnetic line, computing the deviation, and entering it into PID to derive the servo adjustment angle. Speed PID controls rotational velocity by defining a target speed, comparing it to the current speed to determine the difference, and substituting this into PID to calculate the motor duty cycle, thereby modulating rotational speed. Together, the positional and speed PID algorithms enable the robust control of servos and motors to facilitate stable, optimized, intelligent vehicle performance on the track.

The students posed several stepwise questions during the inquiry process.

Q1: How is the PID algorithm applied in the electromagnetic tracing of intelligent vehicles?

Q2: What are the theoretical and mathematical formulas of the PID algorithm?

Q3: Ask ChatGPT to write the code controlling the servo using the PID algorithm.

Q4: Ask ChatGPT how to tune the PID parameters.

For questions Q1 and Q2, ChatGPT clearly outlined critical knowledge as follows. First, the PID controller adjusts vehicle output per error by tuning the proportional, integral, and differential terms - enabling more accurate wire tracking. Second, the PID controller steers the vehicle by controlling the speeds of the left/right wheels in accordance with the error. It also regulates speed for stability and smoothness. Finally, the PID algorithm can be adjusted and optimized per specific control object and requirements—the selection of suitable coefficients, error calculations, and parameter tuning to maximize response and stability.

As noted in Ref. 16, obtaining the best-expected response to choose an appropriate objective function, or determining the optimal weight required for optimizing PID controller design is difficult. Stepwise questioning enabled students to systematically gain knowledge on PID controllers and understand tuning for optimal intelligent vehicle control. We present the results for questions Q3 and Q4 in the following section.

3.2 ChatGPT-generated code for PID controller

Figure 3(a) shows part of the ChatGPT-generated code implementing positional PID control on an STC32 microcontroller, including the acquisition/processing of electromagnetic signals, the implementation of PIDs, and output PWM. The *read_sensor()* function collects the

```c
#define SENSOR_PIN  P1_0   // Electromagnetic induction sensor pin
#define SERVO_PIN   P1_1   // Servo control pin

// PID controller parameters
#define KP  1.0
#define KI  0.1
#define KD  0.01

// PID controller state
typedef struct {
  float setpoint;
  float error;
  float last_error;
  float integral;
  float derivative;
} pid_state_t;

pid_state_t pid_state = { 0, 0, 0, 0, 0 };

// Read electromagnetic induction signal
float read_sensor() {
  return (float)ADC_Read(0) / 4096.0 * 3.3;
}

// Initialize PWM output
void init_pwm() {
  PWM_Init(PWM_FREQ_50HZ, 50, 0, PWM_MODE_EDGE);
}

// Set PWM output duty cycle
void set_pwm_duty_cycle(float duty_cycle) {
  PWM_SetDuty(SERVO_PIN, (unsigned int)(duty_cycle * 10000));
}

// Calculate PID output
float pid_compute(float input) {
  pid_state.error = pid_state.setpoint - input;
  pid_state.integral += pid_state.error;
  pid_state.derivative = pid_state.error - pid_state.last_error;
  pid_state.last_error = pid_state.error;

  float output = KP * pid_state.error + KI * pid_state.integral + KD * pi
d_state.derivative;
  return output;
}
```

(a)

```c
// Define the PID parameters
#define KP 1.0
#define KI 0.1
#define KD 0.01

// Define speed error variables
float error = 0.0;
float last_error = 0.0;
float integral = 0.0;
float derivative = 0.0;

// Define target speed variable
float target_speed = 0.0;

// Define current speed variable
float current_speed = 0.0;

// Define PID output variable
float output = 0.0;

// Define PID calculation function
float PID(float target, float current) {
    // Calculate speed error
    error = target - current;

    // Calculate integral term
    integral += error;

    // Calculate derivative term
    derivative = error - last_error;
    last_error = error;

    // Calculate PID output
    output = KP * error + KI * integral + KD * derivative;

    return output;
}
```

(b)

Fig. 3. (Color online) ChatGPT-generated codes of intelligent vehicle driven by electromagnetics for (a) positional PID control and (b) PID speed control.

electromagnetic signal, *init_pwm()* initializes the PWM output, and *set_pwm_duty_cycle()* sets the PWM duty cycle. The *pid_compute()* function calculates the PID output. In this position control system, an electromagnetic detection device amplifies the signal via an operational amplifier and transmits it to the microcontroller's analogue-to-digital interface. The microcontroller processes the analogue input and feeds it to the position controller. After the calculation, a PWM signal controls the steering angle, keeping the vehicle on the electromagnetic line. In the *main()* function, ADC sampling and PWM initialization are performed first. Then, the electromagnetic signal is continuously sampled to compute the PID output used to set the PWM duty cycle. This positional PID implementation acquires and processes electromagnetic data to regulate steering for accurate line following. The code framework provides a reference for developing closed-loop position control systems.

Figure 3(b) shows part of the ChatGPT-generated code that implements PID speed control with an STC32 microcontroller as the central controller for an electric motor system. The PWM output port and encoder feedback pins are assumed to be connected, and timer initialization is complete. The target speed is set on the basis of current track data. Direction and PWM signals are output to the motor driver through the microcontroller PWM port to control the motor start, stop, forward, and reverse. The connected encoder feeds back the motor speed to the timer input pin. The quadrature decoding of encoder pulses generates the feedback signal input to the control system to obtain the motor speed. The PID speed implementation enables closed-loop control by comparing the target speed with the actual speed to regulate the motor for optimized intelligent vehicle performance.

3.3 Parameter optimization of PID controller

Note that the ChatGPT-generated code requires modification per actual situation. In practice, sensor data must be collected for the current angle and output signals must be sent to the servo controller via hardware interfaces. The students found that the initial PID parameters generated by ChatGPT were not applicable to their experimental setup and that they could not trace the trajectory. ChatGPT suggested three strategies for PID optimization aligning with those in the literature.(12–15)

(1) Use a manual adjustment method to incrementally improve control by tuning the proportionality constant K_p, the integral time constant K_i, and the differential time constant K_d.
(2) Use a trial-and-error method to adjust PID values to approach optimal parameters iteratively.
(3) Implement adaptive PID algorithms, such as genetic algorithms, to tune for better automatic control.(13,14)

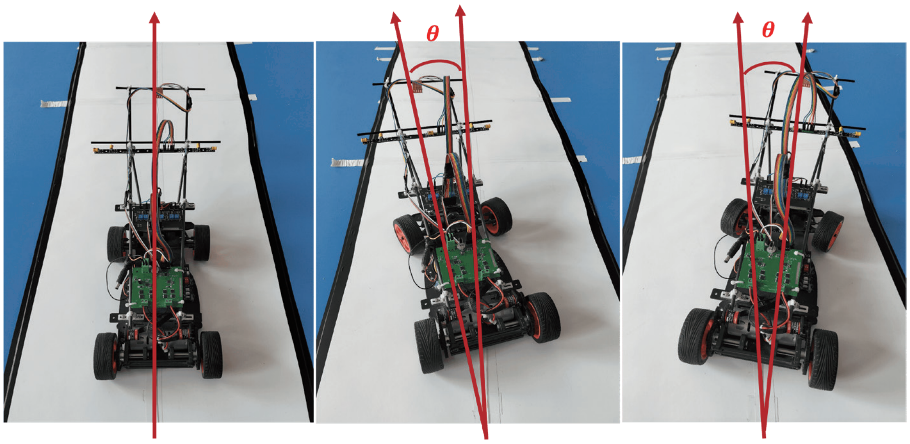
After discussions with the teacher, students combined manual and trial-and-error adjustment methods. The teacher suggested initial values, then the students altered the parameters and observed the effects on the intelligent vehicle. The bounded adaptive output feedback method for student PID tuning proposed in Ref. 16 was used as a guide in this study.

While worthwhile for initial code generation, ChatGPT outputs require modification to make them suitable for real applications. Manual tuning and an iterative approach can help determine optimal PID values for the specific intelligent vehicle and context. Ongoing testing is critical to refine the performance.

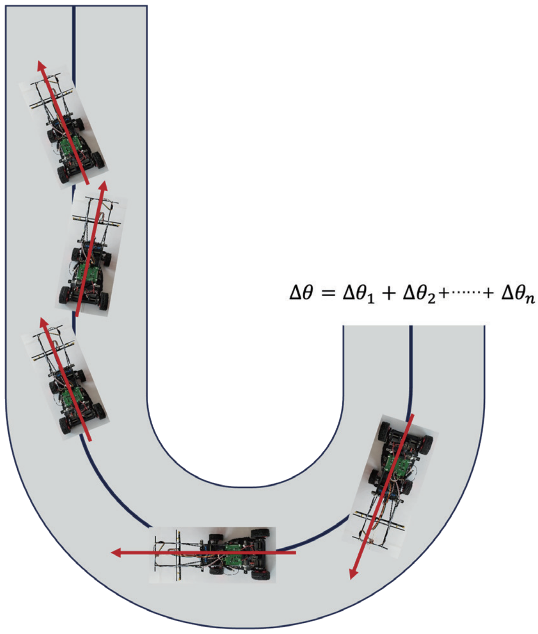
3.3.1 Effect of K_p

When implementing PID control for servo angle regulation in an intelligent car race, the proportional gain K_p is critical in affecting how the vehicle adjusts its angle. With low or zero K_p, the wheels cannot travel straight. Gradually increasing K_p corrects the path. However, an excessively high K_p causes significant front-end oscillation along the line, and as the servo response and angular output become very large, the vehicle swings left and right with each overcorrection [as shown in Fig. 4(a)].

Students consulted with their teacher concerning this "overshooting" behavior. The teacher explained that overshooting can arise from high K_p values, inappropriate K_i settings, or large K_d values and advised the students to reduce these parameters appropriately on the basis of the control object. More profoundly, the instructor provided the insight that a high K_p speeds up the



(a)



(b)

Fig. 4. (Color online) Effects of PID control parameters on intelligent vehicle performance on (a) straightaways and (b) curves with high K_p and large K_i.

response and reduces static error but increases overshoot and reduces stability owing to the instantaneous proportional adjustments. The controller automatically overcompensates if any deviation exists between the set and feedback angles, potentially causing oscillation. Therefore, K_p must balance the trade-off between response speed and stability.

In summary, while ChatGPT provided a starting point to address overshoot, the teacher's guidance offered a more nuanced understanding of balancing PID response, stability, and performance. Ongoing testing and tuning are vital in optimizing control.

3.3.2 Effect of K_i

As shown in Fig. 4(b), a high K_i reduced bias in intelligent vehicle straight-line control, improving performance. However, a severe left–right oscillation occurred after driving straight for some distance following a turn. As mentioned in Ref. 17, this oscillation may be caused by various reasons, among which inappropriate K_i is one.

The instructor provided insight into this occurrence. First, increasing the integral time helps to suppress overshoot, reduce oscillations, and stabilize the system, while too small a K_i decreases stability, increases changes, and leads to cumulative error. Thus, the deviation accumulated by the integral term PID during turning carries over to the straightaway, causing less accurate steering on entering the straightaway. Specifically, although capable of tracking the line, steering is overcompensated and side-to-side oscillation occurs, seriously impacting overall performance.

In summary, while some integral gains can improve straight-line tracking, too many lead to overcorrection. The instructor's guidance helped students to identify and address the root cause of undesired oscillation when transitioning from turns to straights. The fine-tuning of K_i is essential to the optimization of end-to-end control.

3.3.3 Effect of K_d

When implementing PID to control the servo angle, the differential gain K_d critically affects the adjustment of the vehicle's angle. One student set a small K_d, which improved system response and suppressed overshoot to enable faster, smoother turning.

In attempting to enhance further the response by increasing K_d, the student found that performance was degraded by severe oscillations beyond a certain level. This is because differential control is based on the rate of deviation change. Excessive K_d causes overreactions to even minimal deviations, preventing rapid system stabilization.

In summary, the proper differential gain can speed up the response and reduce overshoot. However, too large a K_d will destabilize the system. Determining the optimal K_d requires methodical testing to balance responsiveness and stability. This exercise provided the student with essential information on the nuanced effects of tuning the PID parameters.

4. Knowledge Building in PID Controller Optimization: Lessons from Student Training and Competitions

In this study, three groups of students, namely, Group A, Group B, and Group C, participated in the competition. After two months of training and nearly a month of participation in Shandong Provincial and National Competitions, they acquired valuable professional knowledge and experience. After the competition, we invited students from each group to summarize their insights and expertise from knowledge-building and inquiry-based learning perspectives, as briefly described below.

4.1 PID controller optimization

4.1.1 Knowledge-building perspective

Group A initially relied on superficial judgments and linear control to drive the car but quickly realized that this method was unable to adapt to the varied track environment. Therefore, they used the PID algorithm to optimize the vehicle's navigation system. Through trial and error, they determined the PID parameters, emphasizing that they should be manageable. When adjusting the steering PID control, they noted that too small a K_p would result in slow cornering, while for speed PID control, too large a K_p would cause the car to oscillate.(18–20)

Group B utilized an incremental PID algorithm in their microcontroller and set a detection period of 10 ms. When tuning the steering PID control, they encountered a slow servo response and determined it to be due to too small a K_p. They gradually increased K_p to address this until the servo could turn flexibly. Moreover, they found that too large a K_p caused the motor to oscillate under speed PID control. Thus, they reduced K_p until the oscillation disappeared and multiplied it by 0.6 to 0.8 to determine the final K_p.(17, 21, 22)

Group C proposed various optimization suggestions for the PID control of servo steering. They noted that if the car encounters a large bend on the track or deviates significantly from the target route, the angle of the servo could become very large, causing the front wheel to lock.(16, 23) To address this, they recommended limiting the amplitude of the servo in the program. Moreover, they advised associating the K_p of the servo PID with the processed inductive value to adapt to different bends. They also mentioned that setting a difference ratio and algorithm can achieve track adaptation, especially in situations with many bends.(24) Lastly, they suggested increasing K_d to ensure that the car passes through turns more smoothly.(25)

From a knowledge-building perspective, all three groups demonstrated deep practical inquiry and problem-solving capabilities. Group A's approach was more direct and fundamental, focusing on trial and error. Group B paid more attention to the technical details and precise parameter adjustments. Group C, on the other hand, offered more innovative suggestions, especially regarding adapting to track changes. These results suggest that each group has certain characteristics in knowledge building, but all are striving to deepen their understanding and refine the PID algorithm.

4.1.2 Inquiry-based learning perspective

From the perspective of inquiry-based learning,(2) all three groups showed an active spirit of exploration and autonomous learning. Through constant trial and error, Group A recognized the necessity of the PID algorithm. Group B carried out exploration on a more technical level, continuously adjusting and refining the parameters. Group C was more comprehensive; they considered technical details and proposed various strategies to adapt to different track situations. These results suggest that inquiry-based learning can help students deepen their understanding of knowledge through practice and encourages innovative thinking.

4.2 Code generation via ChatGPT

4.2.1 Knowledge-building perspective

Group A stressed the importance of specifying the platform when using ChatGPT to generate a code. They found that if they did not specify that the program runs on a microcontroller, ChatGPT might produce Python or other non-microcontroller-compatible codes. They also discovered that ChatGPT is highly efficient at spotting syntax errors in C language and can assist in interpreting the code function while providing pertinent comments. This is consistent with the results mentioned in Ref. 7, showing that ChatGPT helps to improve students' programming skills.

Group B emphasized clarity in their requirements when instructing ChatGPT to generate a code. They suggested clearly describing the needs and implementing only a few functions at a time to avoid complications during porting. Furthermore, they underlined the need to validate the ChatGPT-generated code before integrating it into their programs. They also noted that ChatGPT can produce a code in various languages, like C or Python, but sometimes needed to be translated into C. When in doubt about a piece of code, they turned to ChatGPT for interpretation and a summary of its function.

Group C focused on the practical application effects of ChatGPT in programming. They highlighted that ChatGPT has the capability of code completion and generation. It can provide relevant code snippets based on existing code or requirement descriptions, significantly boosting programming efficiency. When using ChatGPT to generate a code, they suggested being precise in requirements and gradually porting the code to ensure stability after each move. They also pointed out that the parameters in the code generated initially by ChatGPT might need to be more precise. However, with actual validation and repeated feedback to ChatGPT, adjustments could be made to achieve more satisfactory results.

From a knowledge-building perspective, all three groups showed deep practical inquiry and problem-solving capabilities when using ChatGPT for code writing. Group A was more concerned with the platform and the adaptability of the code; Group B focused more on the clarity of requirements and the importance of validation, while Group C emphasized the practical effects of using ChatGPT and the significance of repeated verification. This shows that each group has its unique perspective and approach to building knowledge, but all strive to utilize ChatGPT better.

In Ref. 10, it was stated that ChatGPT can generate helpful and personalized feedback and explanations for students, thereby offering them an efficient learning experience. However, note that the efficacy of ChatGPT and other generative AI systems is contingent upon their training data and may unintentionally propagate biases, leading to the generation and dissemination of incorrect information.

4.2.2 Inquired-based learning perspective

From the perspective of inquiry-based learning, all three groups showcased active exploration in applying ChatGPT for code generation. Group A deepened their understanding of the tool by specifying the platform; Group B strengthened the robustness of their code by clarifying requirements and repeated validations; and Group C brought the code closer to actual application through numerous experiments and feedback. These experiences indicate that inquiry-based learning helps students understand a tool's operation and encourages them to engage in continuous experimentation and optimization. As inspired by Ref. 4, integrating other learning models such as analysis, design, development, implementation, evaluation architecture will enhance the outcomes of inquiry-based learning.

5. Conclusions

In this study, we implemented an inquiry-based learning model integrated with ChatGPT for intelligent vehicle competition training. The model promoted the progressive development of problem-solving strategies through student–ChatGPT interactions. Project-based learning enabled collaborative, cross-disciplinary tasks such as vehicle design and algorithm development. Teachers became mentors, co-coaching with ChatGPT to monitor student teams. Students built knowledge with ChatGPT through well-designed prompts, continuously improving acquisition quality and efficacy. This dynamic ChatGPT-powered process stimulated active inquiry and the honing of the problem-solving skill. Integrating inquiry-based, project-based, and teacher-guided learning with ChatGPT helped students gain knowledge, critical thinking, and readiness for competition.

In this study, three groups of students (Group A, Group B, and Group C) participated in a competition after two months of training and almost one month of participation in competitions at both the Shandong provincial and national levels. This intense preparation and participation enriched their professional experience and expertise. Each group had unique insights and techniques, and shared their learnings post-competition.

Group A emphasized the importance of informing ChatGPT about the code's intended platform to ensure compatibility. They found ChatGPT to be efficient in detecting C language syntax errors and invaluable in interpreting and annotating code functionalities. Group B advocated for precise requirement descriptions when using ChatGPT and suggested incremental functionality implementations for seamless code integration. They also highlighted the flexibility of ChatGPT in generating a code in multiple languages with the potential need for translations, especially in C language. Moreover, Group C showcased the practical benefits of ChatGPT, such as code completion. They advised a phased approach to integrating the generated code and iteratively refining it with ChatGPT through real-world testing and feedback.

From a knowledge-building viewpoint, each group's methodology in the competition reflected their distinct approaches to problem-solving. The foundational practice of Group A, the detail-orientated method of Group B, and the innovative techniques of Group C underline the importance of diverse strategies in achieving optimal outcomes. Regarding inquiry-based learning, all groups displayed a proactive exploration spirit. Their shared experiences indicate that such an approach deepens tool comprehension and fosters continuous experimentation and improvement, which are essential for real-world applications. Compared with the research results in Ref. 5, the main advantage of using augmented reality technology (AR) in inquiry-based learning is reinforcing the conceptualization stage. In this study, ChatGPT can comprehensively cover and strengthen various stages of inquiry-based learning.

In this research, machine learning techniques, specifically ChatGPT, were utilized to enhance the learning process in intelligent car racing. ChatGPT serves as a dynamic knowledge repository and takes an active role in the learning journey, providing a rich and interactive environment for students to engage in complex problem-solving activities. The study demonstrates the effectiveness of AI in improving educational experiences and highlights the usefulness of ChatGPT in developing domain-specific expertise.

An AI-powered inquiry-based model equips learners to tackle and solve complex challenges related to intelligent vehicle training and competitions. This approach creates a solid foundation for practical problem-solving that utilizes the strengths of AI. It is important to note that this study involves an inquiry-based learning model that includes learners, ChatGPT, and teachers. In this model, the professional guidance of teachers plays a crucial role in both knowledge-building and the development of professional competence. From this perspective, teacher guidance can also illuminate possible solutions to concerns regarding trustworthy AI issues in education, as described in Refs. 8 and 11.

Acknowledgments

The authors appreciate the financial support of the Shandong Provincial Vocational Education Teaching Reform Research Project (Grant No. 2021164) and Huanggang Normal University (Grant No. 2042022027). They also thank the students Wang Yutan, Yang Runze, Dong Yicheng, Song Wenjie, Liu Jiazhen, He Jun, Chen Ren, Wang Jiayi, and Wu Mingjun for participating in this project.

References

1 National Intelligent Vehicle Competition for College Students: http://www.smartcarrace.com/ (accessed July 2023).
2 M. Pedaste, M. Mäeots, L. A. Siiman, T. D. Jong, S. A. Riesen, E. T. Kamp, C. Manoli, Z. C. Zacharia, and E. Tsourlidaki: Educ. Res. Rev. **14** (2015) 47. https://doi.org/10.1016/j.edurev.2015.02.003
3 S. Verma, M. S. Yacob, and A. Kirpalani: Can. Med. Educ. J. **14** (2022) 89. https://doi.org/10.36834/cmej.75144
4 K. U. Dewi, D. N. Sudjito, and A. Pattiserlihun: J. Inov. Pend. IPA. **8** (2022) 67. https://doi.org/10.21831/jipi.v8i1.43458
5 M. Pedaste, G. Mitt, and T. Jürivete: Educ. Sci. **10** (2020) 94. https://doi.org/10.3390/educsci10040094
6 ChatGPT (We need to talk): https://www.cam.ac.uk/stories/ChatGPT-and-education/ (accessed July 2023).
7 Md. M. Rahman and Y. Watanobe: Appl. Sci. **13** (2023) 5783. https://doi.org/10.3390/app13095783
8 T. Adiguzel, M. H. Kaya, and F. K. Cansu: Contemp. Educ. Technol. **15** (2023) ep429. https://doi.org/10.30935/cedtech/13152
9 D. Baidoo-Anu and L. Owusu Ansah: J. AI **7** (2023) 52. https://doi.org/10.2139/ssrn.4337484
10 J. Qadir: Proc. 2023 IEEE Global Engineering Education Conf. (IEEE, 2023) 1–9.
11 A. Tlili, B. Shehata, M. A. Adarkwah, A. Bozkurt, D. T. Hickey, R. Huang, and B. Agyemang: Smart Learn. Environ. **10** (2023) 15. https://doi.org/10.1186/s40561-023-00237-x
12 R. P. Borase, D. K. Maghade, S. Y. Sondkar, and S. N. Pawar: Int. J. Dyn. Control **9** (2020) 818. https://doi.org/10.1007/s40435-020-00665-4
13 V. Vartika, S. Singh, S. Das, S. S. Mishra, and S. S. Sahu: Advances in Smart Grid Automation and Industry 4.0, M. J. B. Reddy, D. K. Mohanta, D. Kumar, and D. Ghosh, Eds. (Springer, Singapore, 2021) pp. 391–399.
14 V. S. Bhat, V. Kumar, N. Dayanand, A. Shettigar, and Nikhitha: AIP Conf. Proc. **2236** (2020) 080001. https://doi.org/10.1063/5.0006827
15 Y. Li, Q. Chen, H. -C. Chen, and Y. -L. Lin: Proc. 2021 IEEE 4th Int. Conf. Knowledge Innovation and Invention (IEEE, 2021) 34–37.

16 G. Li and K. M. Tsang: Int. J. Control Autom. Syst. **5** (2007) 234.
17 W. Z. Qiao and M. Masaharu: Fuzzy Sets and Syst. **78** (1996) 23. https://doi.org/10.1016/0165-0114(95)00115-8
18 R. Bindu and K. N. Mini: Int. J. Emerg. Technol. Adv. Eng. **2** (2012) 310.
19 B. Aytekin: J. Inf. Sci. Eng. **23** (2007) 1469. https://doi.org/10.6688/JISE.2007.23.5.11
20 H. Wu, W. Su, and Z. Liu: 2014 9th IEEE Industrial Electronics and Applications Conf. (IEEE, 2014) 808–813.
21 J. Liu, P. Zhang, and F. Wang: 2009 Int. Conf. Intelligent Human-Machine Systems and Cybernetics (IEEE, 2009) 206–209.
22 R. Krishnan: Electric Motor Drives: Modeling, Analysis, and Control (Pearson, London, 2001).
23 E. Zergeroglu, W. Dixon, A. Behal, and D. Dawson: Robotica **18** (2000) 171. https://doi.org/10.1017/S0263574799001988
24 Y. Luo and X. Che: 2010 2nd Int. Conf. Advanced Computer Control (IEEE, 2010) 348–351.
25 Z. Sun, R. Wang, Q. Ye, Z. Wei, and B. Yan: IEEE Access **8** (2020) 105031. https://doi.org/10.1109/ACCESS.2020.2994437