Design and Simulation of Self-Driving Cars using Optimal and Machine Learning based control systems.

Author's Name: Ardsher Ahmed.

Coach Name: I-Heng McComb

Host Organization: Stanford

ETP Type: *Classroom*

Subject/Grade: Computer Science / 9-12

Abstract

The Ignited summer fellowship at Stanford enabled me to jumpstart my knowledge and curriculum development towards developing autonomous vehicles control systems. I created a set of lesson plans to walk through and teach high school computer science students to create software simulation environments for real-life road traffic as well as for a self-driving car to navigate terrain in a 2D environment.

Stanford's Autonomous Systems Lab (ASL) is deeply involved in ML (Machine Learning) and AI in robotics and Autonomous control systems for Land, Air, and Spaceborne vehicles. Learning from experts at Stanford I was able to create a set of lesson plans that will enable students to design and simulate autonomous (self-driving) vehicles. With this exercise students will learn the concept of modeling real-life systems in software simulation, learn optimization strategies to minimize the cost and maximize the performance, and understand the concept of ML and its application to optimize trajectory of autonomous vehicles from point A to B along its route.

Focal Content & Supporting Practices

Computer Science Principles. Please see the section "21st Century Skills and Applications" for supporting practices.

21st Century Skills and Applications

Colleges and careers of the future will require students to problem solve and demonstrate the **Four Cs:** collaboration, critical thinking, creativity, and communication. The California computer science standards emphasize these skills.
As a field, computer science itself incorporates problem solving, communication, critical thinking, creativity, and collaboration into its work. The following is a representation of the California computer science core practices and their alignment to equity, problem solving, and the Four Cs.
Core Practice 1: Equity Fostering an Inclusive Computing Culture
Core Practice 2: Collaboration Collaborating Around Computing
Core Practice 3: Problem Solving Recognizing and Defining Computational Problems
Core Practice 4: Critical Thinking Developing and Using Abstractions
Core Practice 5: Creativity Creating Computational Artifacts
Core Practice 6: Creativity Testing and Refining Computational Artifacts
Core Practice 7: Communication Communicating About Computing

During this project all students will have the opportunity to apply the 4Cs on daily basis. For example, as they are working individually they are applying critical thinking and creativity to solve problems, but when they are working together as a group they are not only multiplying their creative and critical-thinking powers but also engaging in communications and collaboration. So, it's a win-win situation to accomplish this goal of practicing 21st Century skills.

Reflection: After the project is completed and presented, students will be prompted to write their reflections on a Padlet about what 21st Century skills they practiced/applied and how they benefited from it during this project.

Measurable Objective(s)

1. Students will be able to create a virtual simulation environment in software to design a virtual self-driving car with limited lookahead capability to detect obstacles in its field of view.
2. Students will create test rigs with static and dynamic obstacles to test drive their car.
3. Students will apply optimization/heuristic techniques to produce an optimal path from start to finish.
4. Students will learn the concept of Machine Learning (ML) in AI and will apply ML techniques to map out an optimal route from starting point to the destination.

Formative Assessment(s)

1. Prior-knowledge assessment about coding, math functions, and geometry.
2. Exit tickets after each lesson.
3. CFUs and real-time feedback on conceptual understanding.

Summative Assessment(s)

1. The simulator code and its execution to demonstrate the simulation of a self-driving car and all related artifacts.
2. Final project presentation.
3. Completion of a graphic organizer on vocabulary, terminology, and concepts in a MS Team assignment.

Fellowship Description

1. **Project:** Design and Simulation of Self-driving Cars using Optimal and Machine Learning based control systems.
2. **Skills:** This project requires skills such as: simulating real-life systems in a virtual environment, algorithm design and analysis, optimization techniques to minimize the cost and maximize the performance, developing ML algorithms and control systems to navigate robots or smart cars.
3. **Careers:** This work relates to the AI software developers and engineers involved in designing the control systems for robots, self-driving cars, unmanned spacecrafts, drones, and other ML based control systems.
4. **Work**: The ASL at Stanford is involved in the development of methodologies for the analysis, design, and control of autonomous systems with a particular emphasis on self-driving cars, autonomous aerospace, and future mobility systems.

Fellowship Connection to School/Classroom

I will use a two-prong approach to share my new experience with my school community.

First, I will share my know-how and research work conducted at Stanford with STEM teachers and colleagues in various PLCs. I will help stimulate a research/educational focus in the STEM subject areas to prepare students to bring them up to speed for the modern day work environment, future technologies, and skills set.

Second, I will show and demonstrate how robots, self-driving cars, satellites, and unmanned spacecrafts are controlled by AI and ML algorithms to perform complex tasks on their own without human intervention!

I will inspire them by showing videos of Autonomous drone racing competitions where humans and machine-controlled drones compete.

With the lesson plans that I have created, my computer science engineering classes will get a taste of real-life engineering problem-solving skills and understanding of a design cycle. They will be able to relate to math, physics, computer science, and material-science concepts as they will implement a virtual simulation of real-life mobile objects. Students will be able to communicate, collaborate, demonstrate, critically-think, be-creative, and be-inclusive of all ideas. The most important of all – they need to be learning and having fun at the same time!

Instructional Plan

Instructional Plan

Students will design a project to develop a virtual road-traffic simulation with multiple moving objects and obstacles. The final goal is to create a self-driving car in the simulator that can autonomously navigate through the traffic as well as the obstacles to go from a starting point to a destination.

Day 0: A Day before starting the new unit, I will give my classes a Prior-Knowledge Assessment to gauge their understanding of new concepts and terminology. This will not only help guide our class discussions but will also help the instructor to fine-tune the instructions.

Day 1: Building the Foundations

Duration	Activity	Details
5 min	Watch a video	To pique student interest in the topic, class will watch a short news video [1] on self-driving taxis and cars that are currently sharing road space in the streets of San Francisco.
5 min	Intro to the unit	Intro to the new unit on autonomous vehicles and self-driving cars. Learning goals and objectives will be covered.
20 min	Front Loading, Direct Instruction, and discussions	Front loading of vocabulary, concepts and CA State Standards (4C's) for all students, especially EL and SPED learners. Direct instructions will be given through PowerPoint slides. Online google translations will also be provided for ELs.
10 min	Group Activity	Groups will search online, discover, and share their findings on various assigned topics and terminologies.
10 min	Exit Ticket	CFU through an exit ticket to gauge their learning.
	HW	MS Teams's assignment on Vocabulary & Concepts Graphic Organizer.

Day 2: Getting your hands dirty by coding for simulation design.

Duration	Activity	Details
10 min	Intro to Simulation	Learning objectives, intro to simulation in MIT's Scratch 3 programming environment [2]. Students will be assigned first activity in starting the project of simulation design and a rubric for grading.
30 min	Do-Now	Students will log on to their computers and go to Scratch programming and start building their simulations as described in PowerPoint (LP1).
10 min	Group Activity	Groups will discuss and share their code with each other and give feedback. They will also demonstrate their work to the teacher.

Day 3: Finish-up & give final touches to your simulations on a short-period day.

Duration	Activity	Details
5 min	LOs	Cover learning objectives, answer questions and issues regarding the coding environment and the project.
25 min	Do-Now	Students will log on to their computers and complete Scratch coding for their simulations as described in PowerPoint (LP2). They will work individually as well in groups to give feedback to each other.
10 min	Exit Ticket	CFU through an exit ticket to gauge their learning and prompt them for next steps.

Day 4: Making the simulation realistic with fast and slow moving vehicles.

Duration	Activity	Details
10 min	Front Loading & Direct Instructions	After covering the learning objectives and goals of lesson 3, students will be prompted to think about realistic road traffic situations with cars traveling at dissimilar speeds. They will discuss in groups to come up with solutions to detect obstacles, such as a slow-moving vehicle in front of a faster car, and how to handle the situation safely.
20 min	Do-Now	Students will log on to their computers and go to Scratch programming and start building on top of the previous day's design work to make cars travel at

		different speeds. They will follow the directions on a PowerPoint (LP3).
10 min	Group Activity	Groups will discuss and share their code with each other and give feedback. They will demonstrate to class how they avoid running into obstacles or slow-moving vehicles. They will discuss multiple solutions and implement the safe and most economical option in their code.
10 min	Exit Ticket	CFU through an exit ticket to gauge their learning and prompt them for next steps to think about.
	HW	Each student will complete and turn-in the code to the teacher for next day's presentations and grading.

Day 5: The Trials & Presentation Day.

Duration	Activity	Details
5 min	LOs	Cover learning objectives of the day and get ready for the presentations.
35 min	Final Project presentations	Each student will run their simulation code on a SmartBoard and demonstrate a 2-lane traffic simulation, and answer any questions. Teacher will grade the projects.
10	Reflections on 4C's	After the class presentations, students will be prompted to write their reflections on a Padlet about what 21st Century skills they practiced/applied and how they benefited from it during this project.

Summary: The sequence of this multi-day lesson plan on designing a simulation project is as follows:

Day 0: Prior-knowledge assessment conducted (LP1: slides 1 to 2).

Day 1: Intro to the topics and covering the fundamentals in LP1 slides 3 to 10, such as: showing a Video, intro to the unit, front-loading of vocabulary, concepts, definitions, and standards, a do-now group activity, an exit ticket for CFU (slide 13), and assigning a graphic organizer.

Day 2: Coding begins for the simulation project in LP1 slides 15 to 20. Students will create all the required artifacts for the simulation project.

Day 3: Coding continues for the simulation project in LP2 slides 1 to 6. Students will run the simulation in which all the cars are going at the same speed. An exit ticket for formative assessment (slide 8) and a project rubric (slide 9) will be explained to make sure all the bases are covered.

Day 4: Last day of coding in LP3 slides 1 to 7. Students will modify the simulation to implement slow and fast moving traffic, debug and test their simulations. An exit ticket for a formative assessment (slide

8) will be given.

Day 5: Project presentations day LP3 slides 9 to 12. Students will demo their projects to the class for a summative assessment. At the conclusion, students will be debriefed and they will reflect on what skills they practiced during the project and share their thoughts on 4C's (slide 13).

Additional Supports

Each student shall have access to the following tools, Apps and Technologies:

- Access to Google translator
- Front loading of complex terminology and vocabulary
- Front loading of coding and programming concepts
- Modeling, demonstrations, and visual images of concepts and tasks.

Materials

- PowerPoint slides - Lesson Plans: LP1, LP2, and LP3
- Prior-Knowledge Assessment
- LP1-Exit Ticket
- LP2-Exit Ticket
- LP3-Exit Ticket
- Vocabulary Graphic Organizer
- Project Rubric
- Reflections Padlet

References

1. CBS News video on "Waymo leading the future of driverless cars and robotaxis." https://www.youtube.com/watch?v=LzTvLoKGRyk

2. MIT's Scratch 3 Programming. https://scratch.mit.edu/

Keywords

Autonomous Vehicle (Self-Driving Car), Simulation, Algorithms, Optimization, Machine Learning (ML), Artificial Intelligence (AI).