Preparing Software Engineers to Develop Robot Systems

Carl Hildebrandt, Meriel von Stein, Trey Woodlief, and Sebastian Elbaum

Goal

Developing a course that would enable upper-level undergraduate students in computational disciplines to gain expertise on foundational aspects of software development for robotics

Impact

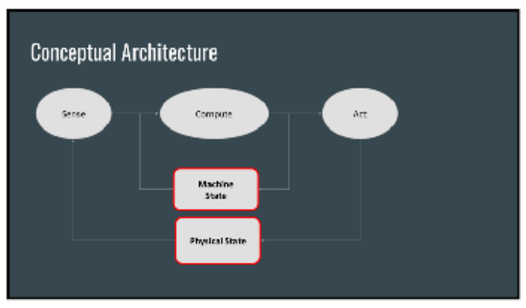
26 Students

4 Placed in Robotics Industry

Guiding Principles

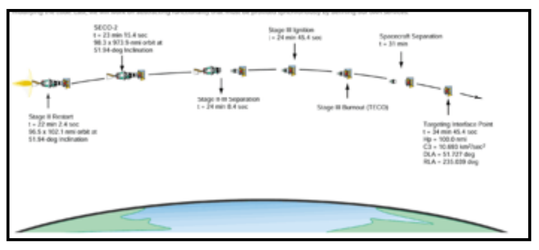
	Principles
P1	Prioritize the challenges of robotics that are unique from other CS systems
P2	Focus on the unique software engineering techniques and practices required by robot system development
P3	Provide opportunities for experiential learning to encourage students to practice and reflect on their experience
P4	Lower adoption barriers by making the material more accessible
P5	Reinforce foundational material across both SE and robotics

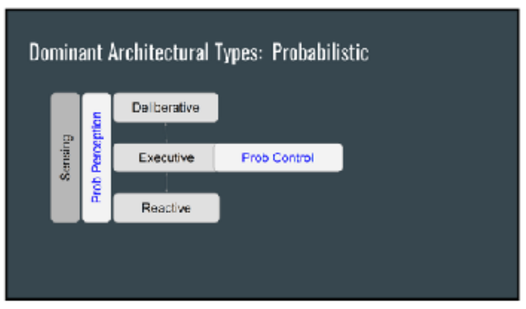
Course Structure - Week 3



Aim: Fundamental concepts

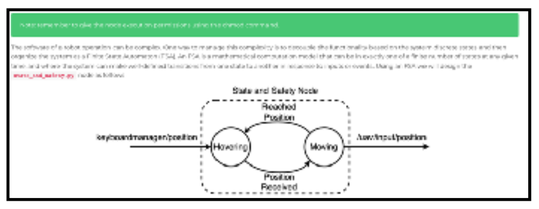
Real world example



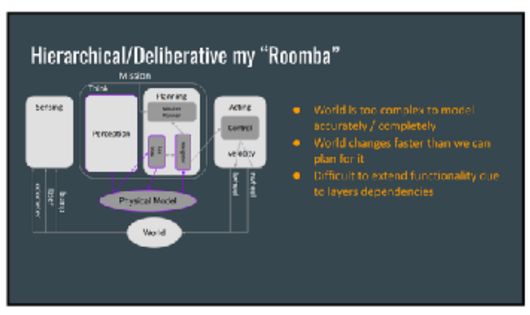


Begins with the basic conceptual architecture of robotics to encode state

Asked to implement a basic FSM

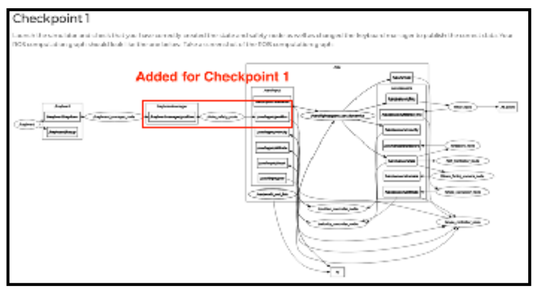


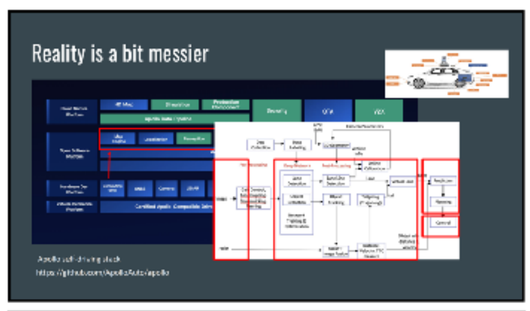
Highlight how to debug code



Covers critical domain-specific architectures

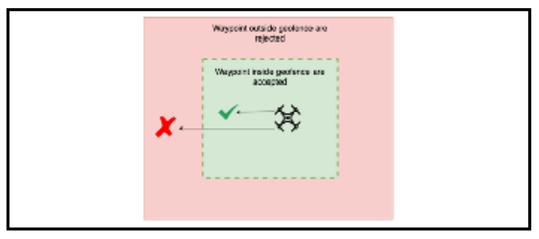
Checkpoint allowing reflection and discussion



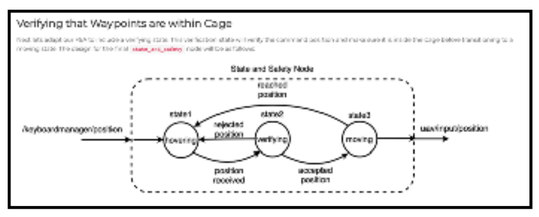


Discuss design tradeoffs over different scenarios

Emphasis developing code that is easily parametrizable



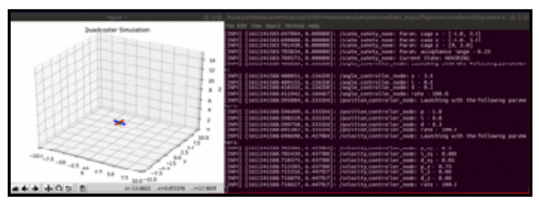
Increase FSM complexity, allowing for more complex behavior



Introduce FSMs

- Represent them in code
- Scaling them to robotics
- How they assist in under standing the real world

Testing through simulation



Getting Started

Lessons Learned

What worked well

1. Pairing SE and robotics topics
2. Building flexibility into the course
3. Using different levels of abstraction
4. Incremental scaffolding of course material
5. Team structure and process
6. Demonstrating and reflecting during checkpoints

What needs improvement

1. Pairing SE and robotics topics
2. Variety of student machines
3. Discussion are time extensive
4. Unclear prerequisites
5. Require an empirical assessment of the success