15-494/694: Cognitive Robotics

Spring 2025

Professor:
Dave Touretzky
dst@cs.cmu.edu

TA:
Kailash Jagadeesh
kailashj@andrew.cmu.edu

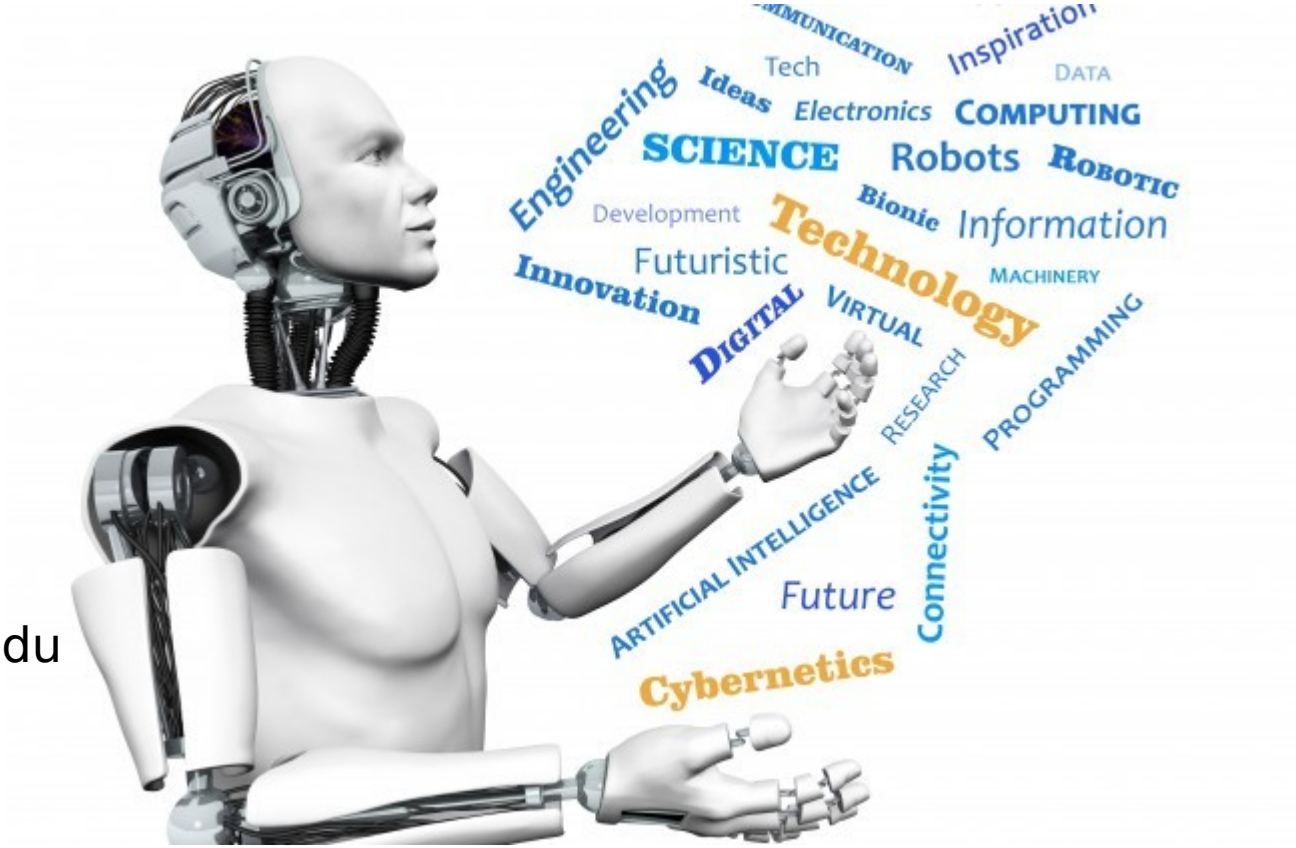


Image from http://www.futuristgerd.com/2015/09/10

What Is This Course About?

- Investigating robot intelligence using the VEX AIM robot and GPT-4.
- Developing robot programming skills:
  - Industrial-strength Python 3
  - Robot vision using OpenCV
  - Machine learning using PyTorch
  - Large language model APIs
- Exploring the “Ten Big Ideas in Robotics”.

Administrative Stuff

- Course Times/Locations:
  - Mon / Wed 4:00 to 4:50 in WeH 5320
  - Fri 3:30 to 4:50 in TEP 1001 (AI Maker Space)

- Course home page:
  http://www.cs.cmu.edu/afs/cs/academic/class/15494-s25

More Administrative Stuff

- Please come to class on time.
- We'll use Piazza for communication. It’s better to ask questions via Piazza instead of emailing the professor or TA directly.
- Assignments will be handed in via Canvas. Assignments are due by 11:59 PM on the day indicated in the schedule.

Grading

- Lab participation (10%)
- In-class quizzes (10%)
- Programming assignments (50%)
- Final project: do something cool (30%)

Difference between 15-494 and 15-694:

- More stringent grading for 694.
- Possibly some extra programming problems.

In-Class Quizzes

- Quizzes test your knowledge of the previous lecture or reading assignment.
- They’re short.
- They’re worth 10% of your grade.
- Be sure to keep up with the reading and review the lecture slides so you’re prepared for the quizzes.

Robotics Education 20 Years Ago

Lego Mindstorms and Vex IQ:

- $250 to $350
- No vision: blind robots are boring
- Unreliable components

Sony AIBO ERS-7:

- $2000 in 2003
- Color camera; program it in C++
- Discontinued in 2006
- Came back in 2018 (can’t program it)

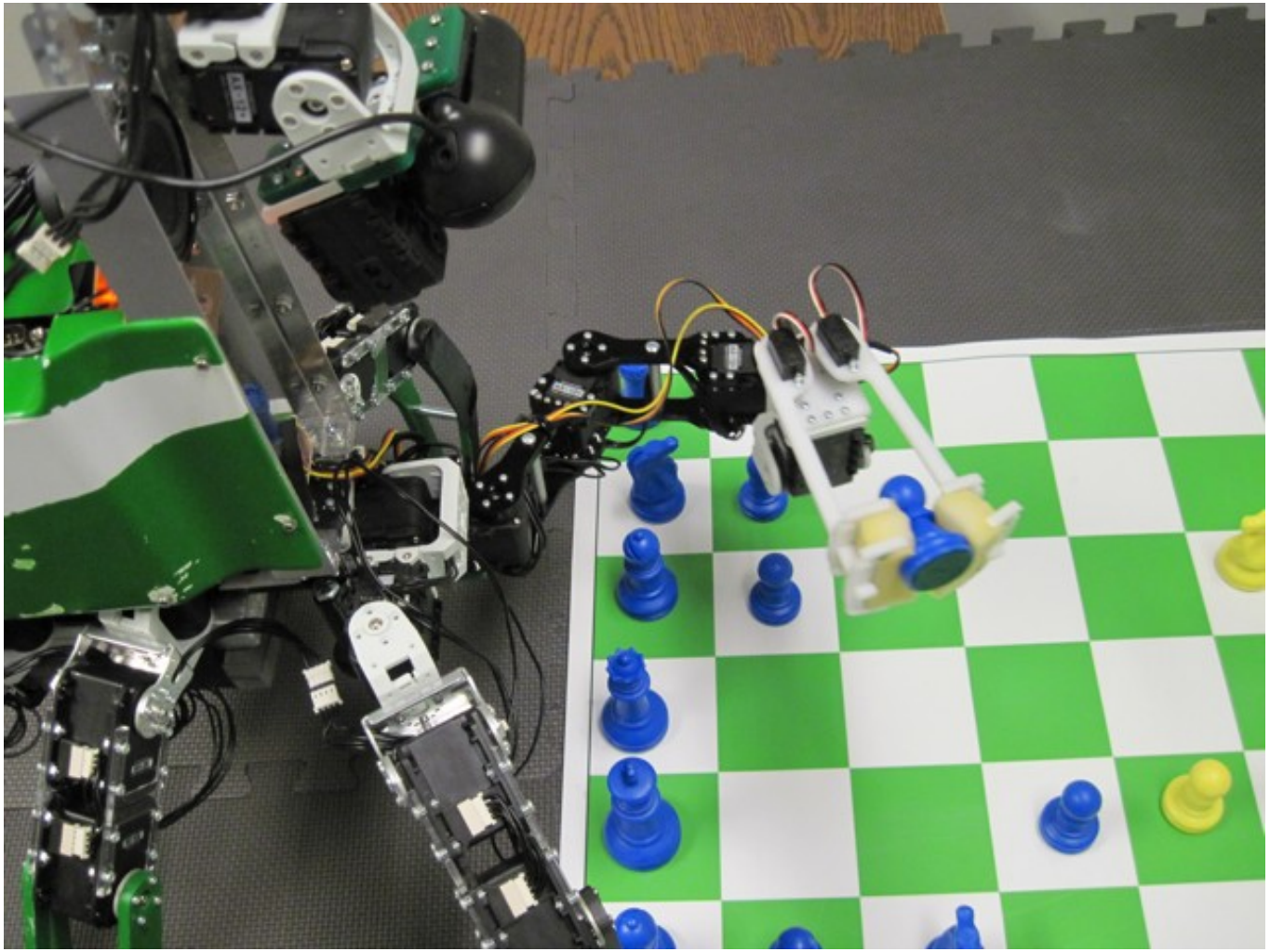
Aldebaran/Softbank's Nao V5 humanoid:

- $9000 in 2017
- Vision, AI algorithms, ROS
- Humanoids are good at two things:
  - Looking cute
  - Falling over

Chiara Playing "*Ode to Joy*"

Demo by high school student Ashwin Iyengar, August 2010.

Chiaras Play Chess at AAAI-2010



Calliope Robots

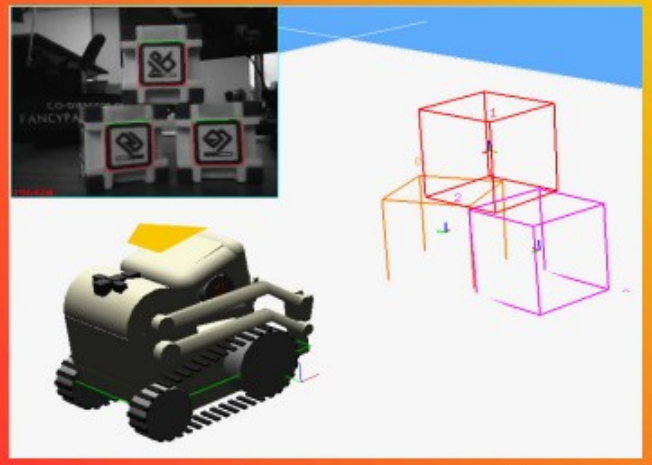
15-494*/694* Cognitive Robotics: The Future of Robot Toys

*Course redesigned for 2017

Cozmo by Anki is a new vision-guided mobile manipulator with built-in artificial intelligence and an open source Python SDK. This course will analyze and program Cozmo.

Topics include:

- Robot software architecture
- Human-robot interaction
- Computer vision
- Navigation
- Path planning
- Manipulation



Spring 2017 – 12 Units
Instructor: Dave Touretzky
Dates/Times: Mon/Wed 3:30 – 4:20
Fri 3:00 – 4:20
No prior robotics experience required, just strong programming skills.

You can get your own Cozmo for $180 at anki.com.

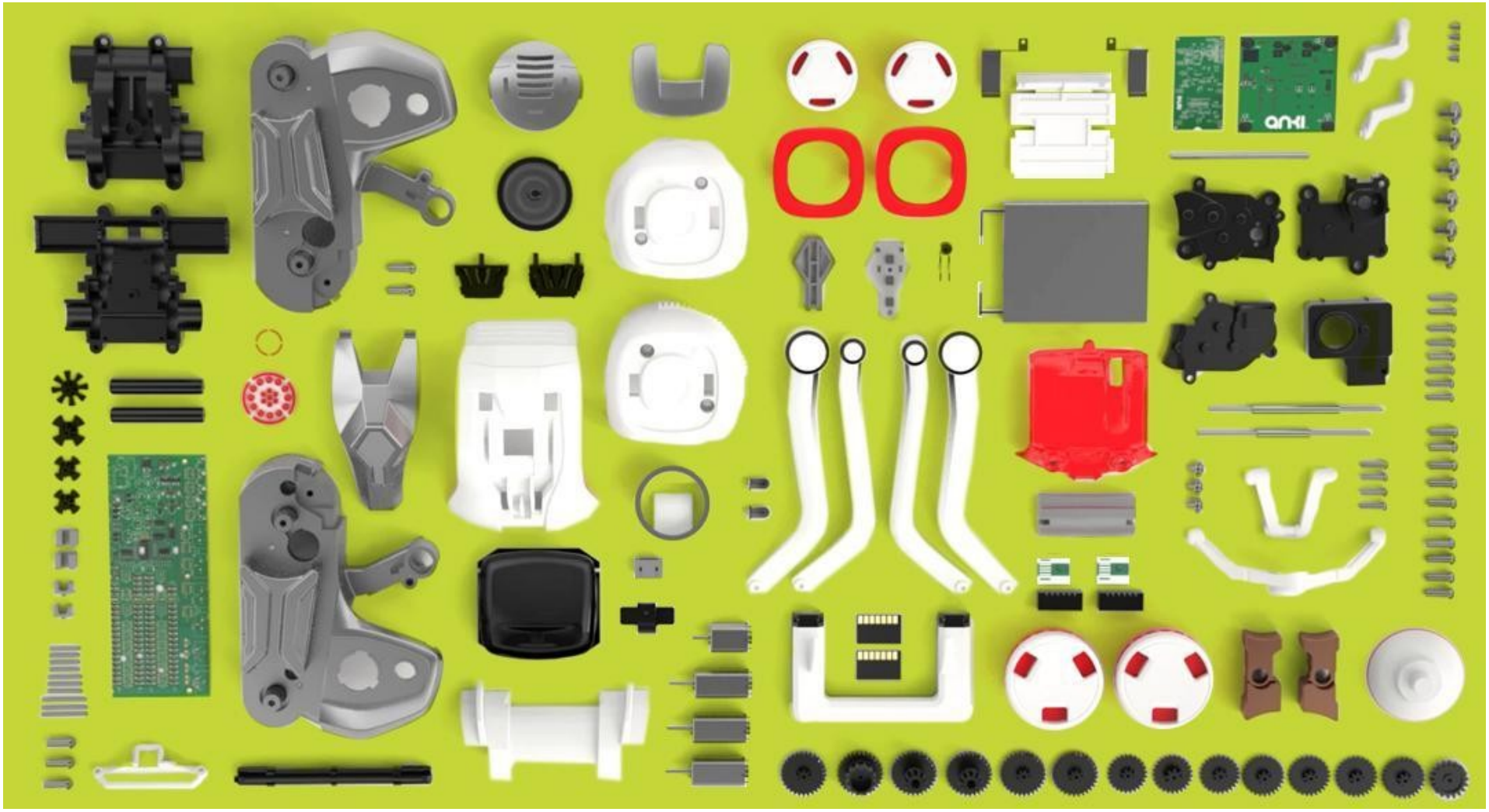
Cozmo Was the Best Platform

- First shipped mid-October 2016.
- Open source Python SDK.
- Anki went bankrupt on May 1, 2019.
- Digital Dream Labs purchased the assets in December 2019, but...
- We need a new platform now.

What Was Cool About Cozmo?

- Cozmo was a **vision-guided mobile manipulator.**
  - Very few consumer robots have vision.
  - It could see special markers, recognize human faces.
- Open source SDK, plus 2 million lines of proprietary code.
- Cozmo used AI algorithms internally.
- Cozmo was ridiculously cheap: $179 at Amazon or Best Buy.

Over 300 Parts



Vector

Vector Specs

- Qualcomm APQ8009 processor on board
- 1080p color camera
- Array of four microphones
- “Time of flight” laser rangefinder
- Four cliff detectors
- Accelerometer and gyroscope
- Touch-sensitive head panel
- Color “face” display
- Software was never finished

New for 2025: VEX AIM

- Color camera (640x480)
- Omni-directional wheels
- Magnetic grip/kicker
- Audio output
- LCD touch screen
- WiFi interface
- Python SDK

Available Q1/Q2 2025 for $179.

VEX AIM

vex-aim-tools

- CMU-developed SDK for VEX AIM built on top of the vendor-supplied SDK.
  - https://github.com/touretzkyds/vex-aim-tools
- Succesor to cozmo-tools, which in turn descended from Tekkotsu.
- Includes GPT-4 interface, speech recognition and generation, augmented computer vision, path planning, and state machine formalism.

Ten “Big Ideas” in Robotics

- “Big ideas” are the key concepts people should learn when approaching a new field (Wiggins and McTighe).
- “Essential questions” are a way of leading people to the big ideas.
- Read the “Ten Big Ideas” paper linked from the class schedule for today.

1. How Do Robots Know What To Do?

Big idea:

- Autonomous robot behaviors are **mechanisms** constructed from carefully designed algorithms and representations.

Underlying technologies:

- Traditional: state machines; event-based architectures.
- New: use LLMs to generate behavior.

2. How Do Robots See the World?

Big idea:

- Robots use sophisticated but imperfect **computer vision algorithms** to deduce real world object representations from arrays of pixels.

Underlying technologies:

- Deep neural networks; face detection algorithms; ArUco markers; much more...

Machine Learning for Robotics

What can we do with ML?

- Apply existing deep neural net models for problems like object recognition.
- Train specialized vision-based controllers for tasks such as lane-keeping (self-driving cars) or skilled manipulation.
- Whatever you can think of!

3. How Do Robots Know Where They Are?

Big idea:

- Robots estimate their position in the world using a combination of **odometry**, **visual landmarks**, and other types of sensor information.

Underlying technologies:

- Kalman filters; particle filters; SLAM (Simultaneous Localization and Mapping) algorithms.

4. How Do Robots Know Where To Go?

Big idea:

- Robots navigate through the world using a **path planner** to search for routes around obstacles to reach their goal.

Underlying technology:

- Path planning algorithms such as wavefront algorithms or RRTs (Rapidly-exploring Random Trees).

5. How Do Robots Control Their Bodies?

Big idea:

- Robots describe their bodies as **kinematic trees** and use **kinematics solvers** to translate between joint angles and body coordinates.

Underlying technologies:

- Kinematic description files; Denavit-Hartenberg conventions; forward and inverse kinematics solvers.

6. What Can We Do When A Robot Becomes Too Complex for One Person to Fully Understand It?

Big idea:

- Robots are complex software systems that employ standard **abstraction** and **software engineering** techniques to **manage complexity.**

Underlying technologies:

- Modular design; coding standards; class libraries; documentation generators.

7. How Do We Calculate the Quantities Needed to Make A Robot Function?

Big idea:

- Geometry, trigonometry, and linear algebra are the **mathematical underpinnings** of much of robotics.

Underlying technologies:

- Software libraries for linear algebra, angular arithmetic, quaternions, etc.

8. How Can Robots Solve Complex Problems?

Big idea:

- Robots use **task planning** to search a space of **world states** to find a path to a goal state.

Underlying technologies:

- Traditional: task planners; domain description languages; plan execution and monitoring architectures.
- New: large language models (sometimes).

9. How Should Robots Behave Around People?

Big idea:

- Successful **human-robot interaction** requires awareness of humans in the environment and adherence to social conventions such as not following too closely.

Underlying technologies:

- Human-tracking tools (e.g., OpenPose); face, gesture, and speech recognition; natural language dialog systems.

10. How Can Robots Work Together?

Big idea:

- **Inter-robot communication** and **multi-robot coordination** algorithms allow robots to collaborate.

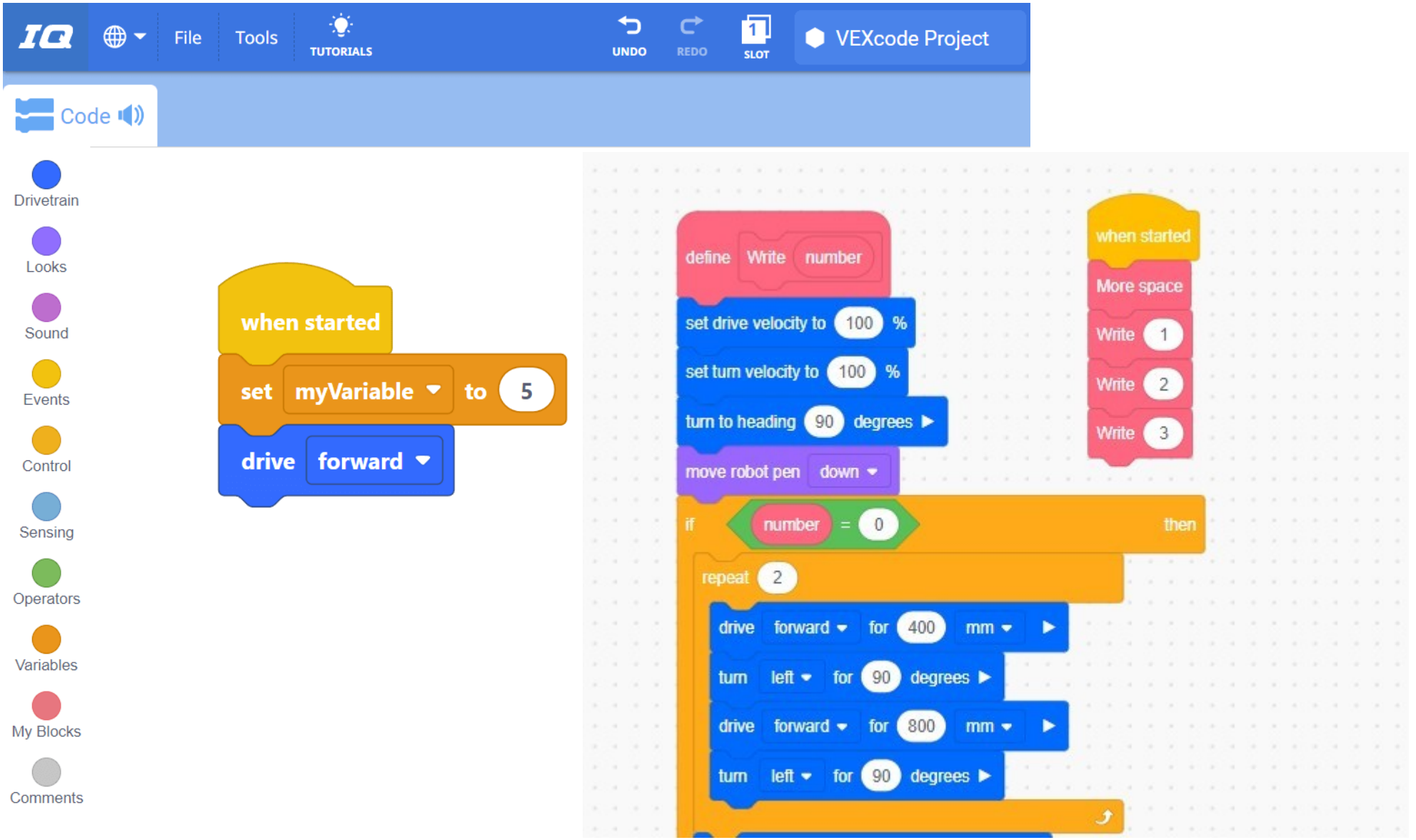
Underlying technologies:

- Communications primitives; shared world maps; multi-robot planners.

Multiple Programming Frameworks for VEX-AIM

- VEXcode uploads to and runs on the robot
  - Blocks-based version looks like Scratch
  - Python version (Python subset)
- VEX AIM Python SDK runs on the laptop
  - vex-aim-tools is built on top of this
- Calypso – runs on a laptop
  - Initially developed for Cozmo
  - Calypso for VEX AIM is in development

VEXcode Blocks Programming



Calypso is “Kodu for Robots”

What You Should Do Now

1. Bookmark the course home page.
2. Read the “Big Ideas” paper.
3. Brush up on your Python 3: classes, modules, numpy.
4. Install vex-aim-tools on your personal machine.
5. Note that Friday labs start at 3:30 PM, not 4:00. Lab attendance is required.