The City College of New York

CSC 36000:
Modern Distributed Computing *NextGen with AI Agents*

By Saptarashmi Bandyopadhyay
Email: sbandyopadhyay@ccny.cuny.edu
Assistant Professor of Computer Science
City College of New York and Graduate Center at City University of New York

Today's Lecture

Communication Models for Distributed Systems

Single-Agent Q-Learning

Multi-Agent Q-Learning

- Independent Q-Learning
- Value Decomposition Networks
- QMIX

Walkthrough on Coding Assignment 1

Important to set up Google Colab to get free compute

- Helps Class Assignments, Projects

Syllabus walkthrough

https://saptab.github.io/modern-distributed-computing-with-AI-Agents/static_files/Syllabus.pdf

Communication Models

Shared Memory

If you're brainstorming for the group project, you might take turns writing on a whiteboard so that everyone on your team can read your awesome idea!

This is what happens in a **Shared Memory** model. Multiple processes (students) can both write to and read from a shared portion of memory (whiteboard).

Shared Memory is flexible and highly efficient, but it runs the risk of *race-conditions* where processes run the risk of overwriting each other, leading to inaccurate calculations.

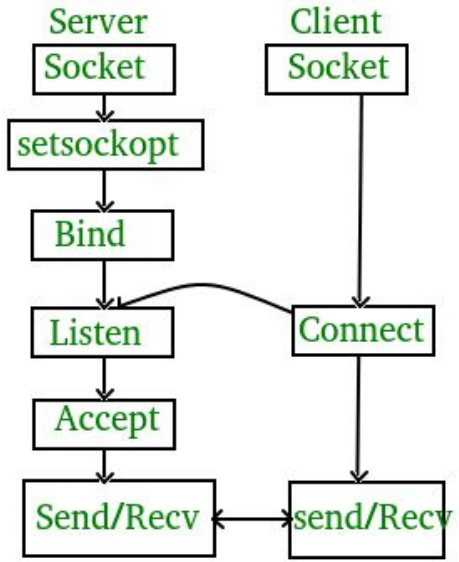
- Race conditions are very hard to debug!

Message Passing

Here, we have no shared state; communication happens through a designated channel, such as a *network socket*.

This takes time, but there's no risk of race conditions, making it ideal for networks of separate devices.

Many common real-world protocols are designed to do message passing, including in the internet (TCP/IP), synchronous systems (REST APIs), and asynchronous queuing systems (MQTT)



Distributed Algorithms for AI Coordination

Introduction to Single-Agent AI

The Coordination Problem

Many real-world problems involve multiple decision-makers (“agents”) that need to work together in order to accomplish a common goal:

- Autonomous Driving
- Energy Grid Management
- Autonomous Drone Search-and-Rescue

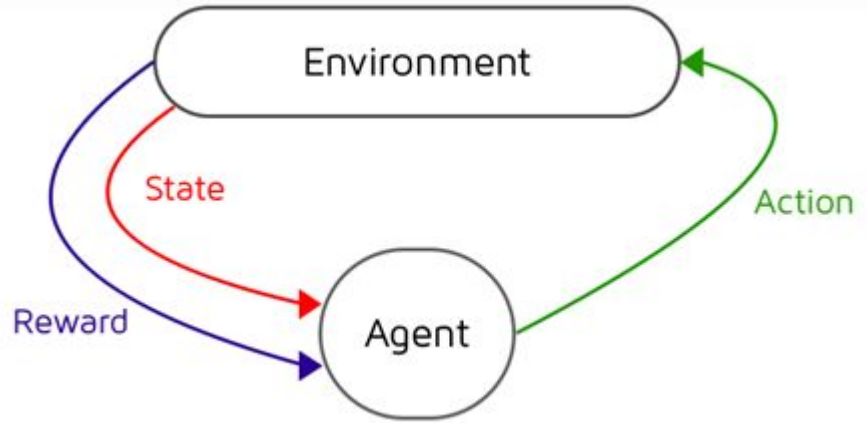
How these agents coordinate is a complex question, so it’s useful to understand how a single agent knows what actions to take.

Single AI Agents

We usually model the world (“environment”) of an Agent as something called a **Markov Decision Process** (MDP).

An MDP Consists of:

- States **S**
- Actions **A**
- Transitions **T**
- Rewards **R**



Questions?