Multi-Agent Systems

Guide to AI and Data Courses · Part II: Branches of AI

Autonomous agents | Communication | Coordination | Negotiation and auctions | Game theory | Neighboring courses

Sercan KÜLCÜ · Giresun University · Computer Engineering

© Sercan Külcü. All rights reserved. · Interactive HTML version of these slides: https://sercankulcu.github.io/files/ai_courses/slides/Chapter_14_Multi_Agent_Systems.html

INTRODUCTION

What Question Does It Ask?

> **How do several autonomous agents decide together or against each other?**

Multi-agent systems study how several **autonomous agents**, each deciding with its own goals and knowledge, cooperate or compete in a shared environment. One agent's best decision is not enough: what the other agents will do must be taken into account, and joint behavior is steered by communication and rules.

At a glance

APPROACH	INPUT	OUTPUT
interacting autonomous agents	agents, goals, protocols	joint behavior, equilibrium

SUCCESS MEASURE	MATH LOAD	DATA NEEDS
social welfare, stability	medium: game theory	little; simulation

A Short History

1950
Nash defines the equilibrium in which each player best-responds to the others.

1980
Smith publishes the Contract Net protocol, which allocates tasks among agents by bidding.

1987
Reynolds introduces boids, a simulation that produces flocking from three simple rules.

1991
Rao and Georgeff formally define the belief-desire-intention (BDI) agent architecture.

1994
Littman proposes Markov games as the framework for multi-agent reinforcement learning.

1997
The first RoboCup robot soccer tournament is held in Nagoya.

THE COURSE

Core Concepts

Autonomous agent
Decides for its own goal without outside control.

BDI architecture
Agent model that reasons with beliefs, desires and intentions.

Communication language
Messages such as propose, request, inform: FIPA ACL.

Coordination
Making agents' actions fit together without conflict.

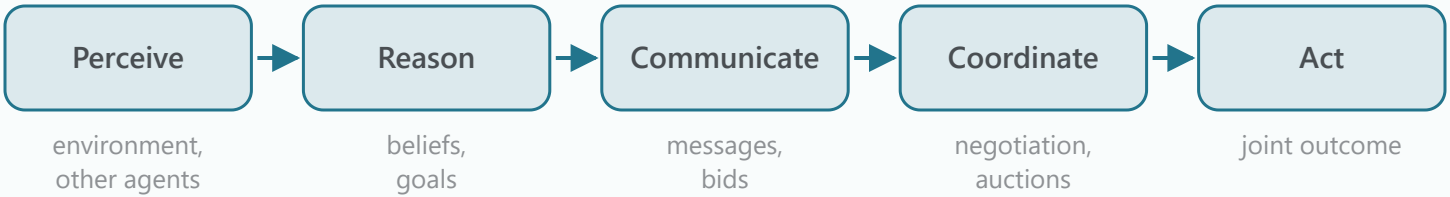
Negotiation
Agents with different interests reaching agreement: bargaining.

Auctions and task allocation
Dividing work with Contract Net and auctions.

Nash equilibrium
No agent wants to change its strategy unilaterally.

Emergent behavior
Global order arising from simple local rules.

A Typical Workflow



Example: A hundred robots pick orders in a warehouse. When a new order arrives, each robot bids based on its position and the best bid wins the job; robots announce their paths to each other and pass through aisles without colliding. There is no all-knowing central planner.

THE COURSE

What Does the Course Cover?

1. Intelligent agents and types of environment
2. Deductive and reactive agent architectures
3. The BDI architecture and practical reasoning
4. Agent communication: speech acts, FIPA ACL
5. Agent platforms: JADE, Jason
6. Task sharing: Contract Net, distributed planning
7. Game theory: Nash equilibrium, prisoner's dilemma
8. Social choice and voting
9. Auctions and mechanism design
10. Bargaining and negotiation protocols
11. Agent-based simulation and emergent behaviour
12. Multi-agent reinforcement learning

Skills you will gain

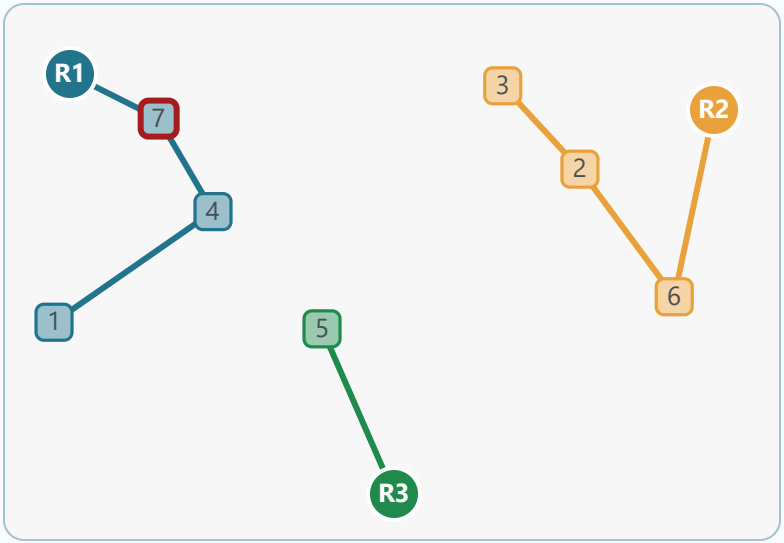
writing agents and messaging in JADE | simulation with NetLogo or Mesa | learning agents in PettingZoo | finding Nash equilibria in small games | designing auction and bargaining protocols

A typical semester; the order and coverage vary between universities.

TRY IT

Delivery Robots Bid for Packages

Three delivery robots work on campus with no manager. Each package is auctioned; each robot bids the extra distance it would add to its route, and the cheapest wins. Run auctions step by step or click the map to add packages.



Reset | One auction | Allocate all

Packages = **7**

Auction for package 7, extra-distance bids:
R1 9 m · R2 346 m · R3 266 m
→ **R1** wins. Packages left: 0

Total distance: **1044 m** with auctions, 1465 m with round-robin. Each robot bid knowing only its own route.

The problem was solved not by one program but by **negotiating agents**: in the contract net a manager announces a task and agents bid according to their own cost. Adding a robot does not break a central plan.

CONTEXT

Prerequisites

Courses worth knowing before this one, and why they are needed.

Course	Why it is needed
Artificial Intelligence	the agent concept, collective decisions
Algorithmic Game Theory	Nash equilibrium, mechanism design and auctions
Object Oriented Programming with Java	objects and messaging: the language of agent platforms such as JADE
Algorithms	graph algorithms and complexity of distributed problems

CONTEXT

Are You Ready for This Course?

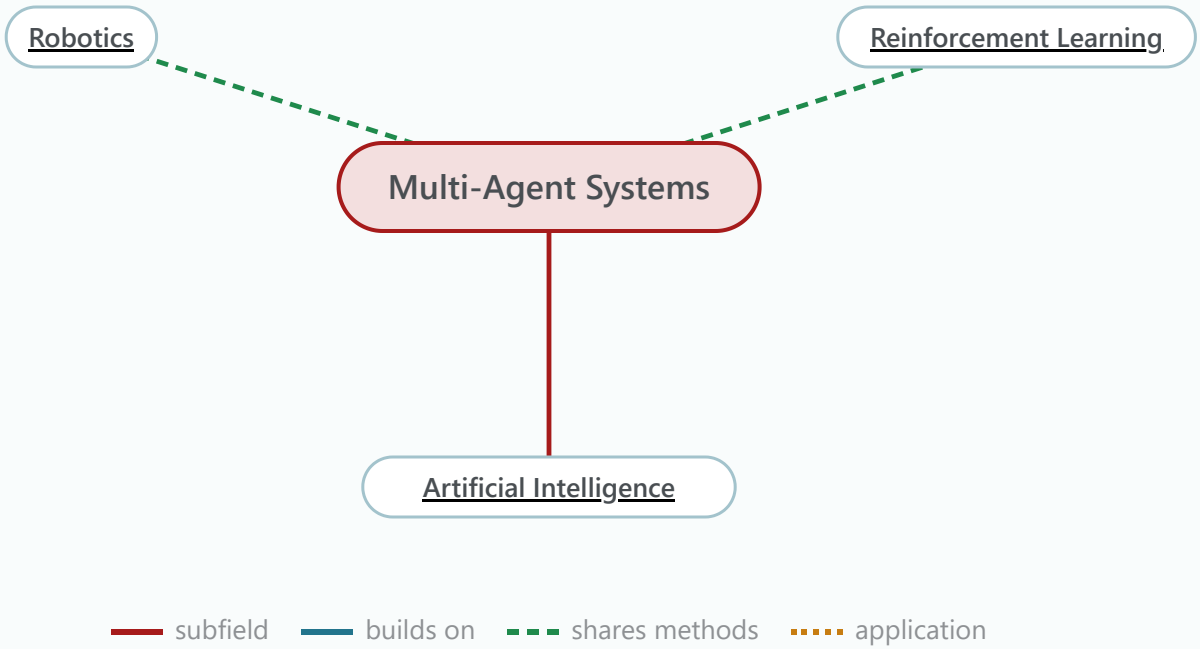
Ask yourself; for every "no", the right-hand column shows where to start.

Ask yourself	If not, start with
Can you find the Nash equilibrium in a two-player payoff matrix?	The opening chapters of an algorithmic game theory course
Can you explain a single agent that searches a state space?	Artificial Intelligence
Can you write classes and have objects send messages via method calls?	An intro to object-oriented programming in Java
Can you write an algorithm that finds the shortest path in a graph?	Graph algorithms from an algorithms course

If you can say "yes" to most of these, you are ready; one or two gaps can be closed during the semester.

CONTEXT

Neighbouring Courses



Overlapping Courses

Relation	Course	Shared ground
subfield	This course is a subfield of Artificial Intelligence	the agent concept, collective decisions
shares methods	Shares methods with Reinforcement Learning	multi-agent reinforcement learning, game theory
shares methods	Shares methods with Robotics and Robot Vision	swarm robotics, coordination

DIFFERENCES

Where It Differs (1/2)

Criterion	Multi-Agent Systems	Artificial Intelligence	Reinforcement Learning
Core question	How do several autonomous agents decide together or against each other?	How does a machine act intelligently by searching, reasoning and planning?	How can good decisions be learned by trial and error from a reward signal alone?
Approach	interacting autonomous agents	search, logic, probability and learning combined	trial and error with rewards
Input	agents, goals, protocols	problem definition: states, actions, goal	environment: state, reward
Output	joint behavior, equilibrium	action sequence, plan or decision	policy: state → action
Success measure	social welfare, stability	reaching the goal, cost, rationality	total expected reward
Math load	medium: game theory	medium: logic, probability, graphs	high: probability, dynamic programming
Data needs	little; simulation	little; often a model and knowledge suffice	generated through interaction

- **Artificial Intelligence:** AI seeks one agent's best decision against its environment; this course studies joint behavior where other agents also decide.
- **Reinforcement Learning:** RL studies one agent learning from reward; this course also covers communication, negotiation and coordination, with or without learning.

Where It Differs (2/2)

Criterion	Multi-Agent Systems	Robotics	Evolutionary Optimization
Core question	How do several autonomous agents decide together or against each other?	How does a machine sense the physical world, know where it is and move?	How do we find a good solution when there is no gradient or the search space is huge?
Approach	interacting autonomous agents	sense-plan-act loop	population-based heuristic search
Input	agents, goals, protocols	sensor measurements	objective, solution encoding
Output	joint behavior, equilibrium	motor commands, path	good, not proven optimal, solution
Success measure	social welfare, stability	task success, safety	quality, evaluation count
Math load	medium: game theory	high: transformations, control	low-medium
Data needs	little; simulation	physical experiments and simulation	none; evaluations suffice

- **Robotics:** Robotics solves sensing and motion for one machine; this course deals with dividing work among many robots or software agents.
- **Evolutionary Optimization:** Swarm intelligence algorithms use a swarm as a tool to solve a problem; here the agents and their interaction are the subject.

PRACTICE

Applications

Warehouse robots
Sharing tasks and paths among hundreds of robots.

Smart grid
Energy trading between producers and consumers.

Traffic management
Coordinating intersections and vehicles.

Automated negotiation
Online bidding and ad auctions.

Games
Computer opponents that play as a team.

Social simulation
Epidemic, market and crowd behavior.

PRACTICE

Risks and Ethics

Unforeseen collective behaviour

Agents that each work correctly can together produce surprises: in the 2010 Flash Crash, automated trading sped up the fall.

Collusion without talking

Pricing agents that learn can, without any agreement, learn to punish each other and settle on prices above the competitive level.

Gaming the mechanism

An agent that finds a loophole in an auction or voting rule can bid or vote untruthfully and tilt the outcome its way.

Mistaking simulation for reality

Results of agent-based epidemic or market simulations depend on the assumed rules and should not go straight into policy.

In depth: Explainable AI and AI Ethics.

PRACTICE

Tools and Datasets

Tools and libraries

JADE | NetLogo | Mesa | PettingZoo | Jason (AgentSpeak) | Repast | MASON

Classic benchmark problems

Prisoner's dilemma | Predator-prey problem | RoboCup soccer | StarCraft Multi-Agent Challenge (SMAC) | Schelling segregation model | Flocking (boids) simulation

In this course data is mostly not collected but **generated by simulation**: agents' rules are written and joint behavior is observed by running them.

PRACTICE

Common Misconceptions

Common belief	In fact
A multi-agent system is a multi-core program.	In a parallel program the parts serve one goal; agents are autonomous and their goals may conflict.
If each agent does its best, the whole is best.	As in the prisoner's dilemma, individually rational choices can lead to an outcome bad for everyone.
Coordination needs a central manager.	Swarms and auction protocols achieve decentralized coordination through local rules and messages.
Agents always cooperate.	Agents can also be rivals; the course studies cooperation, competition and mixtures of both.

NEXT STEPS

After This Course

Courses that build on or extend this one

This course is mostly an end point: it builds on the others.

Related careers

Robotics software engineer | Simulation and modeling specialist | Game AI developer | Logistics systems engineer | Researcher

Multi-agent systems extend AI with a social view. After this course you can move to multi-agent reinforcement learning for learning agents, swarm robotics for physical teams, or algorithmic game theory for designing incentives and rules.

NEXT STEPS

Further Reading

- Wooldridge, *An Introduction to MultiAgent Systems*, 2nd ed., Wiley, 2009.
- Shoham and Leyton-Brown, *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*, Cambridge University Press, 2009.
- Weiss (ed.), *Multiagent Systems*, 2nd ed., MIT Press, 2013.
- Albrecht, Christianos and Schäfer, *Multi-Agent Reinforcement Learning: Foundations and Modern Approaches*, MIT Press, 2024.

The End

Chapter 14: Multi-Agent Systems

← Previous chapter · Map of the field · Next chapter →