Iran University of Science & Technology

School of Computer Engineering

Assignment #4

Multi-agent system

BY:

DR. Naser Mozayani, Fall 2024

Teaching Assistants:

Kalantari

Due: 1404/03/10

Contents

Notes:

1. Submit the answers in a complete PDF file and the code for the questions in the .ipynb format (including the notebook cell outputs) in a compressed file named HW4_StudentID.zip by the specified deadline.
2. A total of **72+48** hours of delay in submitting the answers is allowed across all projects. After that, for each additional day of delay, 10% of the score will be deducted.
3. If a student submits the project earlier than the deadline and achieves 75% of the score, up to 24 hours will be added to their allowable delay time.
4. The maximum delay for submitting each assignment is 5 days, and after 5 days, submission will not be accepted.
5. The exercises must be performed individually, and group participation is not allowed.
6. It is important to note that the explanation of the code and the obtained results must be included in the PDF file. Code without a report will result in a score deduction.
7. The evaluation of the assignment will be based on the correctness of the solution and the completeness and accuracy of the report.
8. Please allocate sufficient time for the assignment and avoid leaving it until the last days.
9. You can ask your questions in the relevant group.

good luck.

Answer the following questions.

a) Types of Value in Auctions (10 point)

I. In some auctions, the value of the item for participants may be based on Common Value or Independent Private Value.
II. Explain these two concepts with one real-world example for each.
III. Additionally, when the value of the item for bidders depends on each other's information (Correlated Values), how does this situation differ from the previous two cases? Provide a specific example for this scenario.

b) Challenges in English Auctions (10 point)

I. English auctions, due to their competitive nature and high transparency, are susceptible to certain forms of abuse:
II. Name two mechanisms of fraud or disruption that can interfere with the normal process of such auctions, and describe each with a realistic scenario (examples should be relevant to today's digital or physical environments).
III. For each case, explain how the issue affects the fairness of the auction.

c) Vulnerabilities of the Vickrey Auction (10 point)

I. Despite advantages like encouraging honesty, the Vickrey Auction can face practical issues:
II. Identify one design flaw in this auction that allows for unhealthy strategic behavior (such as bid manipulation). What is the main cause of this flaw?
III. Propose an innovative solution (other than traditional monitoring systems) to reduce this vulnerability.

Multi-Agent Resource Allocation via Competitive Multi-Round Auctions

Problem Description (70 point):

In this assignment, you are required to design and implement a multi-round auction protocol to allocate n resources among m competing agents. Each agent j can be allocated at most one resource, and each resource i can be assigned to at most one agent. The valuation that agent j has for resource i is given by a matrix V[i][j].

Auction Protocol Rules:

I. In each round, every agent can submit a bid for only one resource.
II. If multiple agents bid for the same resource, the resource is temporarily assigned to the agent with the highest bid.
III. In case of a tie in the highest bids for a resource, the winner should be selected randomly and fairly (equal probability).
IV. Each new bid for a specific resource must be at least ε units higher than the current highest bid on that resource.
V. If an agent fails to submit a bid for a given resource in two consecutive rounds, they are eliminated from bidding for that resource.
VI. The auction ends when no bid changes are observed in a full round (i.e., the state becomes stable).

Implementation Requirements:

- Implement a function random_argmax that randomly selects one index among the maximum values in an array (use np.random.choice).
- Design a mechanism to manage simultaneous competition between agents for the same resource, ensuring strategic behavior.
- Track and output the final allocation of resources and their corresponding prices.
- Run the auction protocol at least 5 times using the same input and compare the outcomes to analyze the impact of randomness.

Example Input:

n = 2 # Number of resources

m = 3 # Number of agents

V = [

[50, 80, 70], # Valuations of agents for resource 1

[60, 90, 30] # Valuations of agents for resource 2

]

ε = 10 # Minimum bid increment

Example Output (Run 1):

Resource 1 → Agent 2 at price 80

Resource 2 → Agent 1 at price 60

Example Output (Run 2):

Resource 1 → Agent 3 at price 70

Resource 2 → Agent 2 at price 90

Notes:

- Please use the Python programming language and write your code inside a Notebook.
- It is recommended to structure your code using Agent and Auction classes to manage agent states, bids, allocation status, and bidding history.
- The auction protocol should be designed to scale gracefully with larger numbers of agents and resources.