

Decentralized Coordination Competing for the best strategy!

Intelligent Agents Course

(Credits to Bruno Alves
Michael Schumacher
Radu Jurca)

The Drawbacks of Centralized Coordination

- Centralized Coordination:
 - Tasks known in advance
 - Company has complete information about the parameters
 - The vehicles blindly follow the orders of the company
- the central plan can be unfair
 - NOT incentive compatible

Decentralized Coordination

- agents negotiate and distribute the tasks among themselves
- create a **market**, where tasks are **sold** to the agents that are the most willing to take them
- the competition leads to an efficient delivery solution

Assumptions

- Agents are self-interested and rational
 - $\text{Utility} = \text{reward} - \text{transportation costs}$
 - agents accept only profitable tasks
 - agents maximize their utility
- Market interaction protocol:
 - tasks are auctioned sequentially (one after another)

Our Setting

- one agent is **ONE company**
- the Auctioning House auctions the tasks to the available agents
 - sequential auction (one task at a time)
- agents bid for tasks
 - bid = requested reward per km to deliver the task.
- every task is allocated to the lowest bidder
 - reward received by the winning bidder = lowest bid (first price auction) x distance (on the shortest path) between pickup and delivery
- **bids are binding**

Placing Bids

- lowest bid = (marginal cost) / distance(pickup,delivery)
- marginal cost depends on the context
 - equals the supplementary cost of delivery

Marginal cost to A_i of task t given a remaining set of tasks T :

$$c_{add}(A_i, t) = cost(A_i, T \cup t) - cost(A_i, T)$$

Marginal Cost Example

- A vehicle V without any task
- Task $T1$:
 - Geneva \rightarrow Lausanne
 - Marginal cost: travelling from the present location to Geneva, and then to Lausanne
- Task $T2$:
 - Lausanne \rightarrow Bern
 - Marginal cost (When V already has $T1$): just the supplementary cost, i.e. travelling from Lausanne to Bern.

Considering the future

- when placing bids, speculate about the future tasks that might be auctioned
 - e.g. you have [T1:G -> L], you might be more likely to accept [T2: B -> Z] because you expect a future task [Ti: L -> B]
- consider the probability distribution of tasks

Considering the other agents

- use the feedback from the previous auctions to derive information about the other competitors
- e.g., you might request a higher price if you see that the other competitors are not very efficient

The Competition

- every group has one agent
- every agent controls a company with two vehicles
- we may introduce several dummy agents to check if your agent is at least performing better than our dummy agents
- the competition consists of several runs (each consists of rounds)
- in every round agents compete against the other agent during auctioning of N tasks
- winner is based on the results of all rounds

One competition run

- The Auctioning House receives a list of tasks (drawn randomly from the common distribution)
- all agents are initialized at random
 - random location for the vehicles
 - random costs per km,
 - random max load, speed (do not matter)
- tasks are auctioned and assigned to the lowest bidders
- the platform verifies that each agent delivers all assigned tasks

Rules

- agents can be disqualified if they win tasks but do not deliver them (in the best case the tasks are being auctioned again).
- No cheating (no stealing information about future tasks, use only information which is given in the signals).
- timeout for computing bids = 5 mins (can vary on a machine, so better have some mechanism to check the time elapsed)
- Competition in pairs/groups, winner from each competes with the other winner until only one left

TO DO

- Implement an agent that can participate in the auctions
- implement a bidding strategy (and plan the delivery of assigned tasks)
- WIN THE COMPETITION!

Deliverable

- Deadline specified on the Moodle
- Report of maximum of 3 pages (explain your bidding strategy)
- 100 points
- reward for the winners – to be announced

