ID2209 – Distributed Artificial Intelligence and Intelligent Agents

Assignment 2 – Negotiation and Communication (FIPA)

*Group 20*

Daniel Zanelli

Mehdi Moshtaghi

23/11/2022

# 1. Description

In this assignment, we were tasked with creating a merch store that works with auction-type sales to willing participants at a festival. Signed merch is sold to the winner of the auction. Here we analyze three different types of auctions selling the same items and one auction type selling different items for an easily comparable scenario.

# 2. How to run

Run GAMA 1.8 and import Assignment2.gaml, Assignment2\_Ch1.gaml and Assignment2\_Ch2.gaml files in a new project. Press “gui\_experiment” to run the simulation of each file. Note that the relevant parameters can be changed through the simulation interface without having to change the code. The main assignment has different parameters as the two following challenges.

# 3. Species

## 3.1 Global

The global attributes that define the simulation run are the minimum and maximum prices accepted for the auctions, as well as the “priceInterval” which describes the ascending or descending price steps. Along with these parameters there are the number of auctioneers and participants to be initialized.

## 3.2. Participant

These agents are the guests to the festival. These agents participate in the auctions after being called by the Initiators for a new auction and will respond to calls for proposals with their respective bids. In the main implementation they are initialized with the amount of money willing to spend in the attribute “my\_bid”, which is initialized based on a global boolean variable **“forceNoBid”**, i.e. if it is “true” then all the participants’ “my\_bid” will be lower than the Initiator min price to force no bid. In challenge 2 they are initialized with a budget that is reset after winning an auction, as well as with the utility attribute for each auction type, where the utility is defined as the difference in price paid and the budget that it had for a product.

## 3.3 Initiator

This agent leads a dutch auction in the main implementation and gives the base structure for the other initiators in challenge 2. They have an attribute “price” which holds the current price the item is being offered in the dutch auction, along with a “final bidder” which determines the winning participant. The rest of the attributes for determining the next communication step in the fipa protocol, either by executing an “inform” at the beginning or end of the auction and a “cfp” for each bidding step in the auction.

## 3.4 DutchInitiator

This agent is equivalent to the one described above, but for the challenge 2 implementation in order to keep consistency between the other auction types. This type of auction communicates a price each round that is being reduced from the “maxPrice” by the “priceInterval” at each bidding step.

**3.5 EnglishInitiator**

This agent holds an english auction starting from the “minPrice” and being increased by the “priceInterval” at each step, receiving a list of interested participants on each round. When the bidding step where no participants have accepted is reached, the auction is over and the winner is the participant with the highest previous bid or none if there were no previous bids.

**3.5 SealedInitiator**

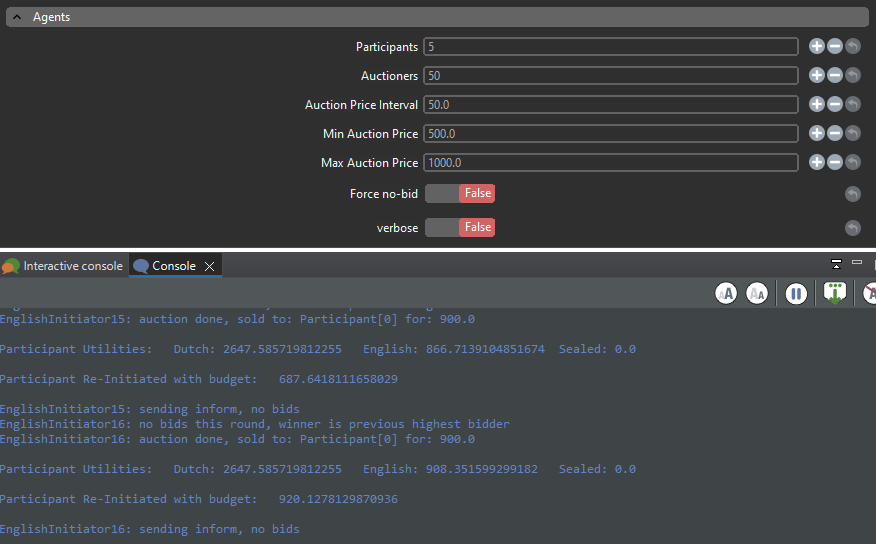
This agent holds a sealed bid that takes each participant's secret bid and considers the winner to be the one who has given the highest bid, if the price is higher than the “minPrice” accepted by the auction. The winning participant pays the full price without being influenced by “priceInterval” or “maxPrice”.

# 4. **Implementation**

We started developing the species Initiator and Participant, where the basic mechanics of fipa communication and dutch auction dynamics were created. Later the code was modified for challenges 1 and 2, where new merch was implemented and two new different auction types were developed.

# 5. Results

In the following image we can observe a typical simulation run with the parameters that can be changed and some terminal outputs showing the outcomes of some english auctions, each participants’ accumulated utilities are shown for different auction types every time they win an auction.

Every sealed auction will always provide 0 utility because the winner will always pay the full budget they are willing to pay for the item, in comparison english and dutch auctions will provide some value since they will almost always pay less than their budget. Some runs dutch auctions give better utility, some english auctions do, while both utilities are on the (approximately) same order of magnitude.

# 6. Challenge 1

To complete the first challenge:

* Global:
  + A hash table “globalAuctionMap” as our initial database is created, containing the info about each merchandise auction.
* Initiator:
  + The type of merchandise is added to the contents of the “cfp” message so that everyone knows from which auction this message is.
* Participant:
  + A hash table “participantAuctionMap” is created to save the desired price for each merch and their respective initiator price.

# 7. Challenge 2

In this challenge we implemented the english and sealed auction types and compared them using a utility function which is intrinsic to each participant. Out of all the initiators, a third of each type of auction is initialized (dutch, english, sealed). Multiple auctions can be done at the same time by the same participants. Each time a winner is announced, their budget is reset to a random value after updating their utility function, each time that happens the participant’s utility and new budget is shown on the terminal.

| *Qualitative/Quantitative questions* | *Answer* |
| --- | --- |
| *Time spent on finding and developing the creative part* | *The creative part took about 12 extra man-hours to conceive and code* |
| In what area is your idea mostly related to… | *Simulation of auctions scenarios where there are either multiple products or multiple auction types* |
| On the scale of 1-5, how much did the extra  feature add to the assignment? | *4. We would say the extra features give an extra layer of depth to the simulation* |
| On the scale of 1-5, how much did you learn from implementing your feature? | *4. The assignment has been a great way to understand the dynamics of auctions and the communication protocol FIPA* |

**8. Discussion / Conclusion**

Creating the ‘Participant’ and ‘Initiator’ species was pretty straightforward, but the assignment got a lot more complex when incorporating different auction types compatible with the same participants and keeping track of individual utilities. We can conclude that the sealed auction is the type that leaves participants with the lowest utility compared to english and dutch auction types. Also, we can conclude that FIPA is a great method for communicating between multiple agents simultaneously.