Introduction

The primary goal of this project is to explore and demonstrate the capabilities of Large Language Model (LLM) based agents within multiagent systems. Specifically, we have simulated a Dutch auction, a type of auction where the price starts high and drops until a participant accepts the price, to evaluate these capabilities. The simulation is executed using OpenAI's language model and features one auctioneer along with three participating agents.

Approach

Communication Structure

The communication within this simulation adheres to a simple yet effective structure. Initially, the auctioneer agent broadcasts a message to all participants. Following this, each participant agent sequentially responds, a process facilitated by single-threaded operation. These responses are collected and managed by a helper manager agent which organises the flow of information back to the auctioneer agent for further action.

Communication Act

The Auctioneer Agent is restricted to responding in the following template, here is an example.

<name>: 'John'
<message>: 'The auction will start at {amount}
<ask price>: ${amount}

Participants are programmed to respond in one of two predefined manners to maintain consistency and predictability within the simulation:

<name>: Not placing a bid
OR
<name>: Placing a bid of ${amount}

This structured approach is essential to prevent simulation divergence due to unpredictable input variations.

Negotiation Protocol

The protocol mirrors the traditional Dutch auction format. If no bids are placed, the auctioneer lowers the asking price. This process repeats until a bid is placed, upon which the item is sold to the bidder offering the highest price at that moment.

Results

For a detailed examination of the simulation outcomes, please refer to the ipynb file available in my github repository.

Reflection

Initially, the simulation was attempted using Langchain, but inconsistencies in their documentation and agent module outputs led to challenges. A switch to OpenAI’s chat completion module significantly improved the simulation's consistency and reliability