**Assignment 3 - Coordination and Utility**

Group 15

Zhinan Gao

Jinyao Zhou

3 Dec 2023

In this assignment, we tackled two distinct tasks. Task 1 revolved around addressing the N Queens Problem, where the goal was to strategically place N chess queens on an NxN chessboard. The challenge was to find a configuration in which no queens could attack each other, ensuring that no two queens shared the same row, column, or diagonal.

Task 2 delved into the realm of festival simulation, drawing connections to previous assignments. The primary focus was on simulating festival stages hosting various shows, with guests making decisions on which stage to attend. The decision-making process was influenced by the guests' preferences and a calculated utility for each stage. This task aimed to capture the dynamics of a festival environment, incorporating the intricate choices made by guests based on their preferences and the perceived value of each stage.

1.How to run

Run GAMA 1.9.2 and import the submission. To run the basic simulation, run festival.gaml. Press main to run the simulation.

**Task 1**

2. Species

Queen:

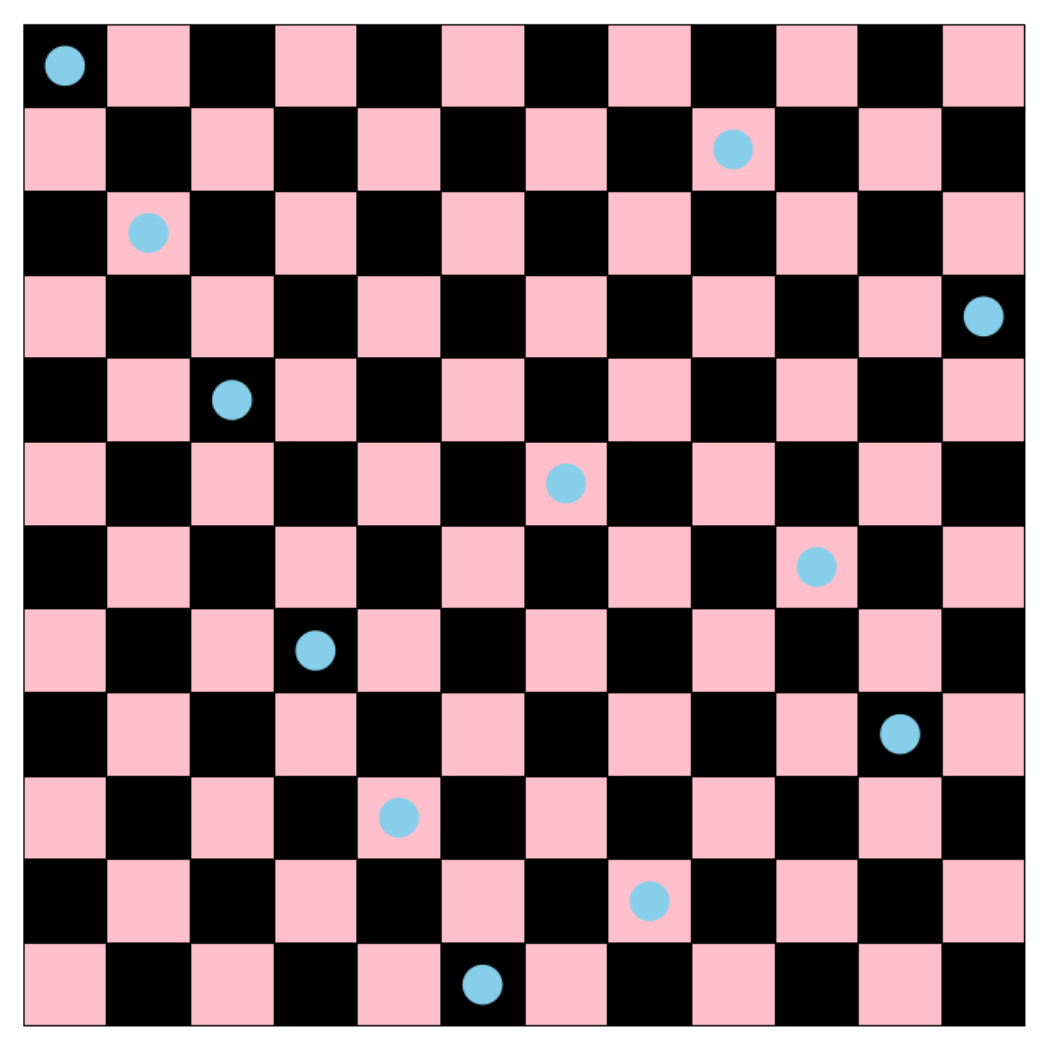
In this simulation, Queen agents are initially positioned along one side of the chessboard and are sequentially added to a list. This list serves as a tracking mechanism for each Queen's predecessor and successor. Once all Queens are initialized, the first Queen in the list initiates the position-finding process. Upon placement, it informs its successor to relocate using the FIPA protocol. If the successor identifies a safe position, it, in turn, informs its own successor to move, and this sequence continues. If a safe position is not found, the Queen communicates with its predecessor, prompting it to move until a secure position is identified.

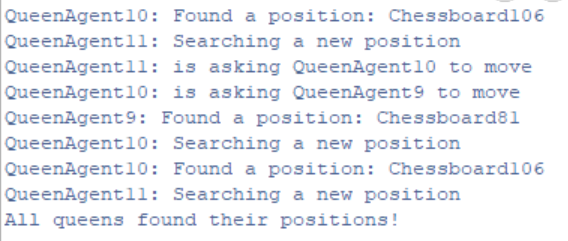
Each Queen systematically searches for a position by traversing all cells from the first row to the last, ensuring that no other queens occupy the same row, column, or diagonal. A matrix, mirroring the size of the chessboard, is utilized to keep track of the positions of all queens. Occupied cells on the chessboard are marked as true in the matrix, while unoccupied cells are marked as false. This dynamic process ensures that each Queen finds a secure position, minimizing the risk of attack from other queens on the chessboard. The simulation concludes when all queens have successfully located their positions or when a safe position is unattainable.

1. Implementation

The implementation of the N-Queens Problem simulation involves a comprehensive model that orchestrates the strategic placement of queens on a chessboard. The global section initializes the chessboard and the list of QueenAgent species, each equipped with FIPA communication skills. The QueenAgent species manages the dynamic process of finding suitable positions for each queen, ensuring they avoid conflicts with one another. The reflexes within each queen govern the interaction protocol, with specialized reflexes handling the initiation of position finding, repositioning requests, and the acknowledgment of successfully securing a position. A dedicated reflex, searchPosition, navigates through potential rows to identify safe locations based on boolean methods validating rows, columns, and diagonals. The chessboard grid is designed with alternating colors to represent the board visually. The entire system is encapsulated in a GUI experiment, offering a graphical representation of the chessboard populated with queen agents, each denoted by a sky-blue circle, signifying their positions. The implementation showcases a dynamic and visually intuitive simulation of the N-Queens Problem, demonstrating the strategic intelligence of the queen agents in securing non-attacking positions on the chessboard.

1. Results





**Task 2**

1. species

1.1 Guest Species:

The Guest species represents individuals attending an event with skills in FIPA communication and movement. They have various preferences, such as lightshow, speakers, and interactivity, and dynamically select acts based on received information from stages. The agent changes its watching state, moves towards the chosen act, and is visually represented by a circle with changing colors.

1.2 Stage Species:

The Stage species simulates event stages with FIPA skills. Each stage hosts an act, and it responds to queries about its acts. Stages automatically restart with a new act if the current act has expired. The visual representation of a stage is a hexagon of color black.

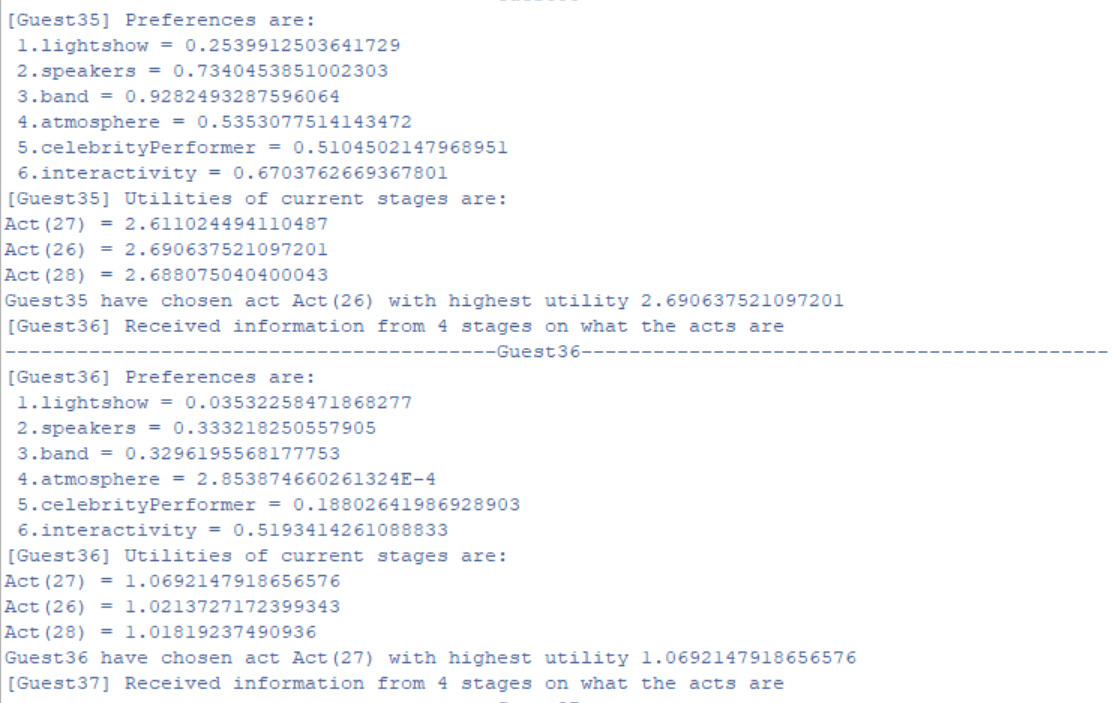
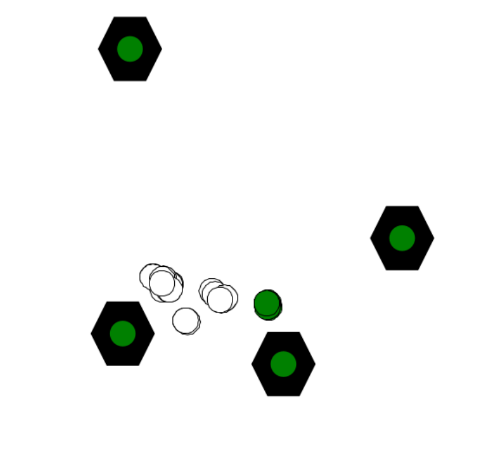
1.3 Act Species:

The Act species represents individual performances occurring on stages. Acts have various attributes, including lightshow, speakers, and interactivity. They have a defined expiration time and are associated with a stage. The visual representation of an act is a circle colored green. Acts are created with random attributes and dynamically assigned to stages.

2.implementation

We implements a multi-agent simulation for a festival scenario where guests interact with stages hosting various acts. The global section initializes the number of people and stages, creating guest and stage entities. The Guest species is designed with preferences for different aspects of acts, and the decision-making process involves FIPA communication to inquire about acts, calculating utilities based on preferences, and selecting the act with the highest utility. Stages, represented by the Stage species, continuously host acts with dynamically generated attributes. FIPA-like communication ensures the exchange of information between guests and stages. The model emphasizes utility-based decision-making, considering factors beyond just music. The GUI output visualizes the entities and their interactions. Overall, the code provides a structured and modular implementation for simulating an interactive festival environment, capturing the dynamic nature of guest preferences and stage attributes.

3.Results



**Discussion/Conclusion**

Extending FIPA from our prior work, we applied it to coordinate a dynamic festival simulation. The model integrates FIPA communication for seamless interaction between guests and stages, enabling dynamic information exchange. Guests inquire about acts through FIPA-like interactions, fostering dynamic information flow. Decision-making involves calculating utilities based on individual preferences, allowing nuanced selections. The model effectively simulates coordination in a festival, showcasing FIPA's versatility for real-time information exchange and decision-making, enhancing realism through dynamic act management and a graphical interface.