

**ID2209 - Distributed Artificial Intelligence and Intelligent Agents**

## **Assignment 3 – Coordination and Utility**

**Group 12**

**Alexandros Nicolaou**

**Alexandre Justo Miro**

**27 November 2019**

## Coordination

In this assignment, we were tasked with completing two tasks:

1. Positioning the speakers of the festival using the N queens problem approach.
2. Visiting acts according to utility function.

## How to run

Run GAMA 1.8 and import 'Basic Model Task 1.gaml' as a new project. Press 'BasicModelTask1' to run the simulation. The same applies for the 'Basic Model Task 2.gaml', 'Challenge Model.gaml' and 'Creative Model.gaml', with the corresponding buttons 'BasicModelTask2.gaml', 'ChallengeModel' and 'CreativeModel'. Note that for Task 1, you only can change the parameter 'number\_of\_queens'; and for Task 2, you can only change the parameter 'number\_of\_guests'.

## Task 1 - Species

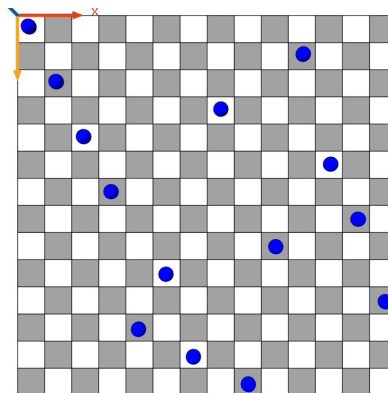
- **Grid:** The grid in that case represents the chessboard in a classical chess problem. The number of columns (and thus rows) is given by the admin, and is being assigned to each of the queens.
- **Queens:** In order for the speakers of the festival to be placed correctly, they are represented as the queens in the N-queen problem. Each queen is assigned a column and can only communicate via FIPA protocol with her predecessor or successor. Using this communication, we first move the first queen to an available position, and recursively move the next one until there is no other possible location. In that case we backtrack and ask for the previous queen to find a new location. This is being done until the problem is solved.

## Task 1 - Implementation

- We created a grey-white square grid with the number of the queens equal to the number of cells per side. We assigned each queen to be able to move in one column.
- By forwarding and backtracking, using the FIPA protocol only in-between two queens we were able to recursively assign a row, and thus a grid position for each queen.
- When a position satisfied all of the constraints, then the queen was assigned that position. Otherwise, when no possible solution was found, based on the positions of the previous queens, then the previous queen had to find a new position.

## Task 1 - Results

The N queens problem solved for N = 14 queens.



## Task 2 - Species

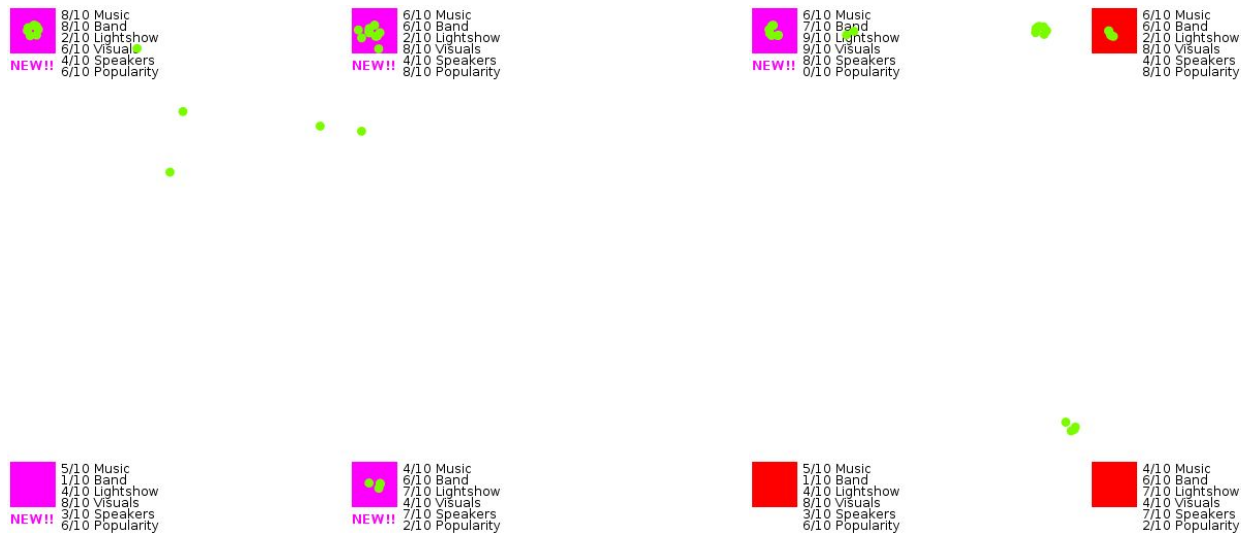
- **Stage:** This agent starts communication with the Guests that are present in the festival. They have a fixed location, and they host acts that last a limited amount of time. Immediately after one act finishes, another act starts. Their only skills are FIPA.
- **Guest:** Guests choose one Stage depending on which act's attributes are more important to them, thanks to messages from Stages received via FIPA. Skills are FIPA and MOVING.

## Task 2 - Implementation

- We created 4 stages which host acts consisting of 6 attributes (music, band, lightshow, visuals, speakers, popularity) valued with an integer between 0 and 10.
- Each stage, by using FIPA protocol, communicates all guests which are the attributes of the act they host. This communication is triggered again each time a new act pops up.
- Each guest receives the data from all stages and decides to go to the stage hosting the act that maximizes its individual utility, according to their particular weights.

## Task 2 - Results

Each time a new act pops up, each guest reevaluates its utility function and decides.



## Challenge

We added the variable crowd mass, and now guests decide where to go according to the maximum global utility instead of their particular individual utility. Where they choose to go now has an influence on the individual utilities of all guests, since there will be one more person in one stage and one less person in another stage, so the crowd masses and therefore the global utility are affected. To coordinate all these movements is difficult, but thanks to a "cloud" where all guests send information and where all guests can receive information from it is not that hard to do. The cases of "staying here" and the three "change to stage" possibilities are evaluated. Obviously, individual utility is sacrificed if changing stage, since guests choose initially the stage that maximizes individual utility. But if summing all individual utilities (including the own one which is sacrificed) yields a higher value, then that guest will choose to change.

## Creative Implementation

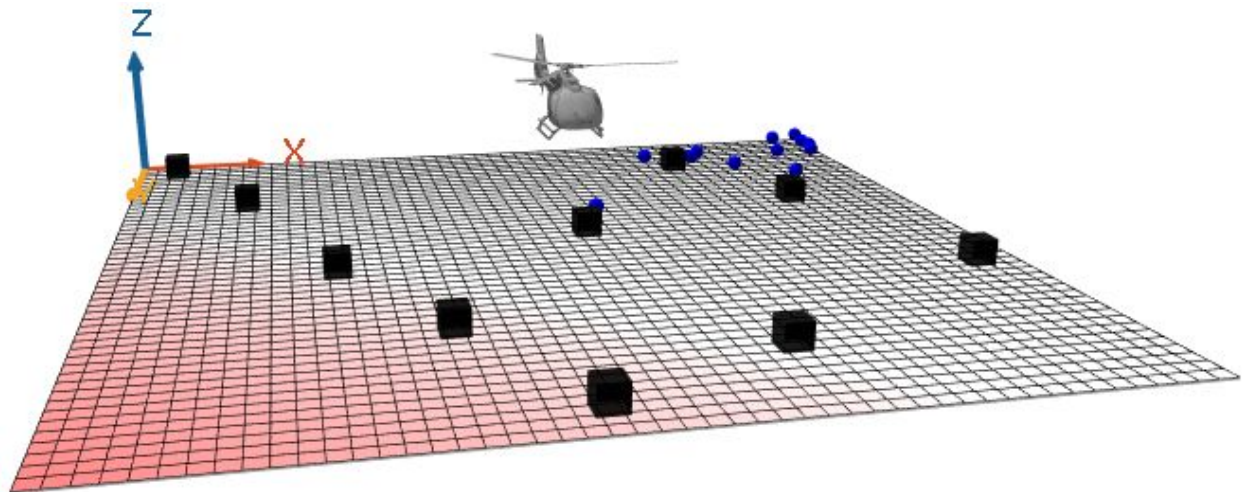
Reaching the third and last assignment of the festival-DAI-series of assignments, we placed the speakers correctly, we set up the stages, and invited artists to entertain our crowd. All seem to run smooth. However, we always have to be prepared for unexpected situations with one of these, being also dangerous and deadly: fire.

After the speakers were placed correctly and our guests started dancing, a cigarette butt was the reason for a fire in the festival area.

In these situations, people start to panic, however there is always one guy, probably security or member of the staff, who is calmer and can safely guide the people to the safest exit.

In order to do that, the rest of the crowd communicate with him via fipa in order to know his location, and started following him.

At the same time, and while the fire spreads all over the place, a 3D helicopter started its engine in order to help take off the fire.



<i>Qualitative / Quantitative questions</i>	<i>Answer</i>
Time spent on finding and developing the creative part	13 hours
In what area is your idea mostly related to...	Fire
On the scale of 1-5, how much did the extra feature add to the assignment?	5
On the scale of 1-5, how much did you learn from implementing your feature?	5

## Discussion / Conclusion

Now that we already know from the previous assignment how FIPA works, we went one step beyond and used it for coordination. The fact that all guests transfer information to the “cloud” or “leader” makes it way easier for the assignment to be solved, since coordinate every guest individually, in an “everyone-by-themselves” approach, would have been messier.