Project Report

ID2209–Distributed Artificial Intelligence and Intelligent Agents

Group 11
Alexander Carlsson - alecarls@kth.se
Abyel Tesfay - abyel@kth.se
2021-11-26

Introduction

In this project we have put all knowledge and experiences we have gained throughout the course labs together in order to put together another view of the various festival simulations we earlier have developed. With no clear instructions to follow this time around we needed to use our own creativity to set the scene while still achieving a set amount of requirements. These requirements follow:

- Create at least 5 different types of guests.
- Each guest type has at least 1 different set of rules on how they interact with other types.
- They also have at least 3 personal traits that affect these rules.
- They have at least 2 different types of places where agents can meet. (Roaming not included.)
- Use at least 50 guests in your simulation.
- Make simulation continuously running.
- Agent communication with FIPA for long distance messaging.
- Have at least 1 global and interesting value to monitor and display on a chart
- At least 1 useful and informative graph.
- Draw out at least 1 interesting conclusion from the created simulation.
    - Example: All agents have some sort of happiness value, ranging from bad (0) to good (1). Show that by adding/removing/changing behaviour of agents how happiness changes over time, to better or worse!

With these requirements in mind we set out to create our project.

Approach

While the requirements presented were not too clear we set out to get started. We decided to make the project in the style of the earlier labs using GAMA and having a two dimensional simulation representing the festival grounds. We looked over all requirements in order to get a scope on what we could develop and made a plan on what to do. We created a basic guest species with the ability to idle, wander and move to targets. We also added a basic place species which the guest could visit. With basic functionality in place we started to add functionality in order to meet all set requirements. How this was done is represented in the rest of this section.

Requirement 1

The first requirement stated that we needed to have 5 different types of guests. In order to fulfill this requirement we imagined a festival where football fans were present and interacted with each other. Our five different types of guests would therefore be guests with 5 different affiliations to football teams. These teams ended up being the swedish teams Allmänna Idrottsklubben (AIK), Djurgården Idrottsförening (DIF), Hammarby Idrottsförening (HIF), Malmö Fotbollförening (MFF) and Idrottsföreningen Kamraterna Göteborg (IFK).

These types were implemented by a string *team* which had the team name of their affiliation. A global list of these teams was created and the guests would pick one team at random on creation. This fulfilled the first requirement.

Requirement 2

The second requirement stated that each guest type needed at least 1 different set of rules on how they interact with other types. In order to satisfy this requirement we needed to specify how our guests would interact. We decided that our guests would simply converse and bond with each other. Our general rule set was made so that guests would simply stay and continue to talk with each other or go talk to someone else depending on if they liked you or not. With a basic rule set we could start the foundations of our unique rule sets. We set up a circular relationship of the team affiliates so that certain teams really disliked certain other teams. These relations are represented in Table 1.

Table 1: table of relations

from\to	AIK	DIF	HIF	MFF	IFK
AIK		dislikes			
DIF			dislikes		
HIF				dislikes	
MFF					dislikes
IFK	dislikes				

Table 1 shows that an affiliate with AIK will dislike an affiliate of DIF and apply a unique rule set on such an interaction. An affiliate of DIF does however not have anything against an affiliate of AIK, but only dislikes affiliates of HIF etc.

With a two dimensional list representing these relations we set up an action to translate team names into indexes and then we had all we needed to start implementing our unique rule sets. This was achieved with one action for all unique interactions and a switch case based on the team string which makes all different guest types gain access to their unique rule set. With these unique rule sets in place the requirement was met. These unique interactions are described below:

AIK

Affiliates of AIK were given the unique rule set to start fights with affiliates of DIF. A fight would occur if the two met in a place and not while roaming. If a fight in a place breaks out all other guests in the place will leave in panic and their happiness will decrease. The fight was made to last for 100 cycles before both the fighters get kicked out of the place. An example of this interaction is shown in Figure 1.

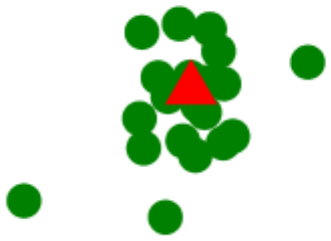

Figure 1: A fight has broken out in a campsite(red). Guests(green) are seen fleeing the scene

DIF

Affiliates of DIF were given the unique rule set to leave the place instantly if any affiliate of HIF was spoken to. This would be in comparison to a regular bad interaction where the guest will just try to speak to someone else in the current place. An example of this interaction in action is displayed in Figure 2.

Figure 2: A guest is seen leaving a bar due to a bad interaction (marked in blue)

HIF

Affiliates of HIF were given the unique rule set to call all their friends if any affiliate of MFF was spoken to. The guest species was given a list of guests only used by HIF affiliates for this purpose. While the list starts off empty, any HIF affiliate who has a good interaction with another HIF affiliate will add that guest to his friends list. If the active guest has no friends and attempts to call his friends his happiness will be affected negatively. An example of this functionality in action is displayed in Figure 3.

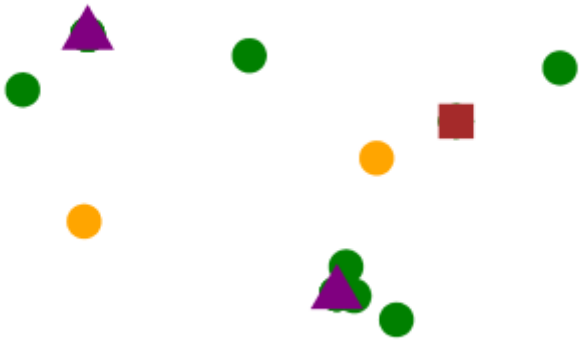

Figure 3: HIF supporters(orange) are on their way to a campsite(bottom triangle) with the goal to help a friend.

MFF

Affiliates of MFF were given the unique rule set to go outside and fight if any affiliate of IFK was spoken to. The two guests will leave the current place and stand together outside and be locked in a fight for 100 cycles before they both leave the place. This only affects the two fighting guests' happiness negatively. An example of this functionality in action is displayed in Figure 4.

Figure 4: An MFF supporter and his opponent(both in cyan) are fighting outside a bar(brown)

IFK

Affiliates of IFK were given the unique rule set of making fun of all affiliates of AIK present in the place if any affiliate of AIK is spoken two. This negatively affects the happiness of all affiliates of AIK present in the place.

Requirement 3

The third requirement stated that we would need to implement at least 3 personal traits that would affect the rules. We added three variables *knowledge, personality* and *alcoholTolerence* which were used in a utility function. This became a central part in the decision if an interaction would be good or bad. It can also affect the unique rules in that a positive enough utility can override the hate that exists between two teams.

As the communication between two guests is local, as in they are both in the same place, this is done using an ask clause. As with earlier utility labs guests also have preferences for the 3 personal traits. These are multiplied and added together.

Requirement 4

The fourth requirement was to have at least 2 types of places where guests could meet. We made a basic place species and added string for representing the two different types bars, and campsites. These two types were made to be represented with different shapes and colors but function mostly the same. Guests can meet and interact. When a guest goes to a place the guest is added to a guest list of the place and when the guest leaves he is removed. This list is available for other guests during interactions. Places were made to be initiated locally without a type and are later on set to the different types in the global initialisation.

The interactions between guests were made to be affected by the type of place. In bars guests interact as usual but in campsites guests are less likely to have bad interactions. This can be since most guests are tired when going back to their tents, there is a lack of lights so they can not see each other etc. This was implemented as a bonus value to the calculated utility functions if an interaction takes place in a campsite.

Requirements 5, 6 & 7

Requirements 5, 6 and 7 are smaller requirements which we had most functionality for already. The ability to have at least 50 guests present in the simulation was easily fixed by modifying a global variable *numberOfGuests* which specified the number of guests, something that would satisfy requirement 5.

As we already had guests leaving places and resetting variables in order to get the urge to go visit another place and interact with more guests, we also already had a simulation that was continuously running. This was also easily achieved since we had no way for agents to die during the simulation. This satisfied requirement 6.

Finally to satisfy requirement 7 which needed FIPA messaging for long range communication we made sure that guests affiliated with HIF used FIPA to call their friends when they needed help at a place. This is the only long range communication that is done, and with FIPA applied to it the 7th requirement was satisfied.

Requirements 8 & 9

Requirements 8 and 9 cover the needs of a global interesting value and the monitoring of said value with at least 1 informative chart. We implemented one global universal happiness variable which serves as a shared pool of happiness for all guests in the simulation. If one guest has a good time, the universal value is increased by one, if two guests get into a fight, the universal value is decreased by two etc.

The changes of this global happiness value was implemented to all interactions where it made sense. Some events do not have direct effects on the universal happiness, but instead have indirect effects. An example of this is the HIF call for friends case. While no changes to the happiness are made as all friends arrive, we have an indirect effect as a place has been filled with all HIF affiliates.

We implemented a graph which follows the tracked value with margins of 100 above and below the tracked value. The graph also shows 100 before and after the current time. An example of this graph is provided in Figure 5.

Figure 5: Example of graph tracking universal happiness

Requirements 10

The final requirement stated that we were to draw one interesting conclusion from the simulation. When all other requirements were met we started to run and analyze the simulation deeper than before. During development and testing we sparked a few ideas of what could be interesting to test later. The interesting conclusion and the experiments used to draw this conclusion can be found in section *Experiments and Results.*

Experiments and Results

Some changes were made in order to properly run a couple of experiments. Firstly we implemented an even distribution of team affiliations that would remain constant between experiments instead of teams being assigned randomly at startup. This would help to get better results when studying the results of all experiments.

With the aim of studying the universal happiness to draw conclusions on what affects it, the experiments started by trying to establish a baseline of what would be considered “normal” behaviour in order to see how we could affect the behaviour.

With 5 teams and 10 guests of each team, 4 bars and 4 campsites, 10 executions were made where the simulation was allowed to run for 30 000 cycles. The results of these executions followed one pattern. All 10 executions had an in general increase in global happiness. The charts for the 10 executions are displayed in Figure 6. Observe that these charts have the same x-axis scale of 30 000 while the y-axis differs based on the maximum value reached for each execution.

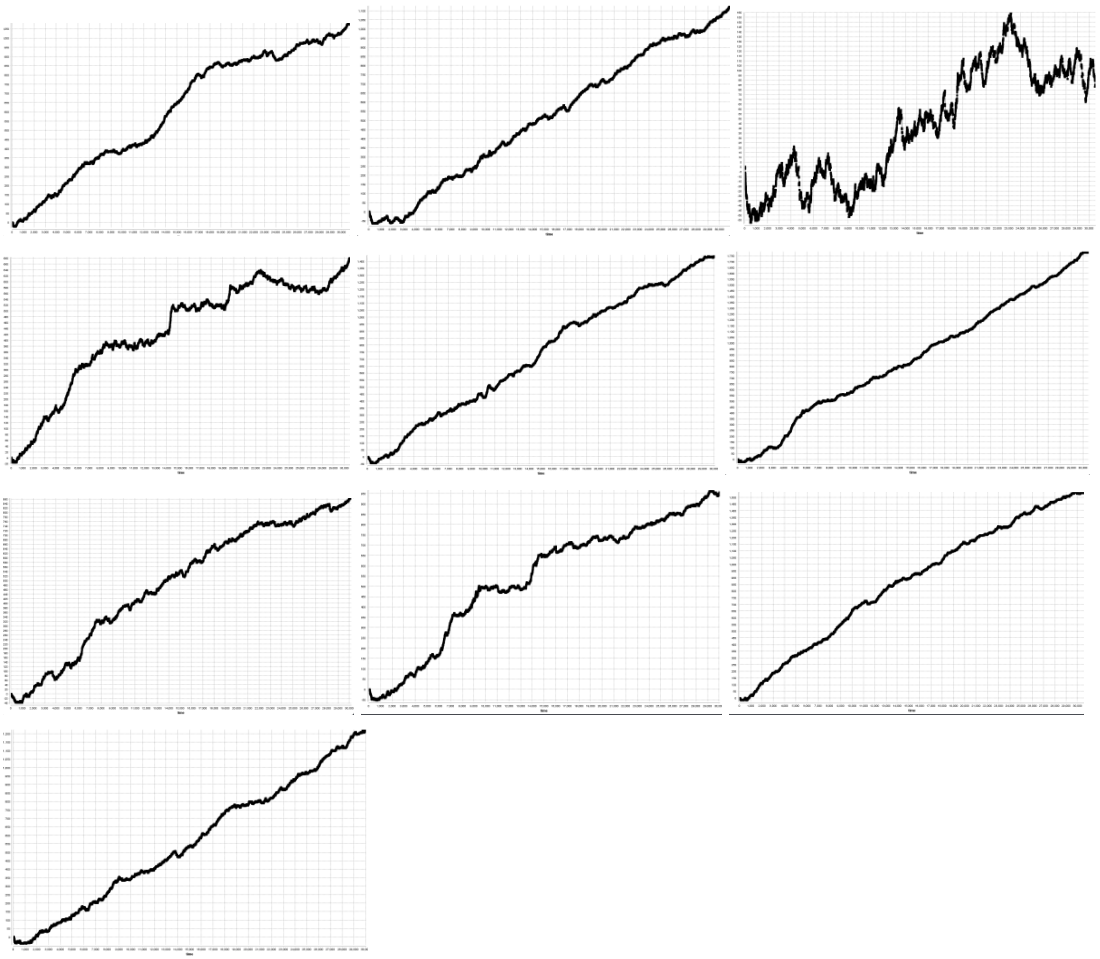

Figure 6: The 10 first executions

In order to analyse what affects the global happiness we started by changing one parameter at the time. We started by reducing the amount of campsites and increasing the amount of

bars in the simulation. While keeping the total amount of places to 8 we reduced the amount of campsites to 1 and the bars to 7. This resulted in more variation in the simulation charts as the patterns were a mix of rising, decreasing and mixed charts. Some examples from 6 executions are provided in Figure 7.

Figure 7: Executions with changed place distribution

These results indicate that the base interactions, purely based on the utility functions and team interactions, are fairly evenly distributed. By providing an increase in campsites for guests to visit we get an almost always increasing global happiness pattern. The same results can be achieved by changing the behaviour of all guests so that they less likely visit campsites, or straight up ignore them. This behaviour can also be achieved by removing the utility bonus gained from campsites.

This shows that campsites play a big role in providing an increase in global happiness. This is not too strange as by adding to the calculated utility we do not only prevent bad interactions, but we also prevent team-unique interactions where effects are further impacted.

In order to further study the impact of places to visit and team-unique interactions we decreased the total number of places while keeping the distribution even. We reduced the amount of bars to 1 and campsites to 1. While the places were reduced all other values were kept as default as in the executions in Figure 6. This resulted in only decreasing happiness charts in contrast to the execution in Figure 6. These decreasing results are displayed in Figure 8.

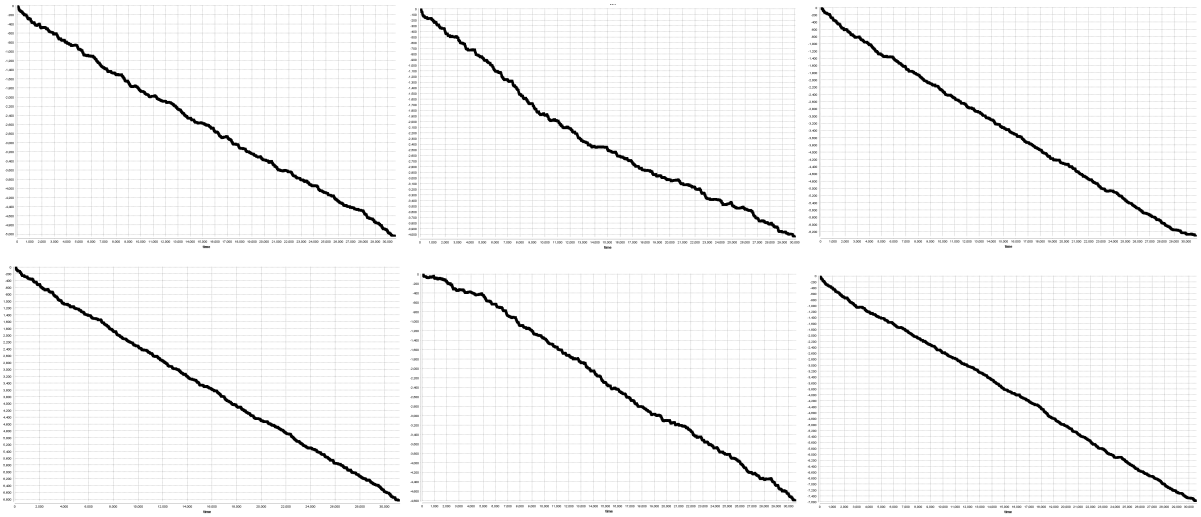

Figure 8: Executions with only 2 places

This behaviour is actually to be expected. With fewer places to go, more people of different teams get cramped together and we have a higher chance of team-unique interactions. We also have a much larger effect from certain team-unique interactions. Affiliates of AIK affect everyone in a place negatively when they start a fight. With fewer places to go and higher populations in the 2 remaining places increases the effect of this interaction.

Discussion and Conclusion

The experiments we ran provided us with plenty of interesting results from which we can draw a couple of interesting conclusions. While some values of the simulation still are randomized during each experiment, contributing with some uncertainty, patterns shown are still considered to be rather clear.

Some interesting conclusions drawn from the created simulation and experiments follow:

Number of places heavily affect global happiness

The total number of places play a big role in the pattern of global happiness development. By increasing or decreasing the amount of places guests can visit, global happiness development is affected as the total amount of interactions themselves are increased or decreased.

Campsites heavily affect global happiness increase

As campsites give a bonus to the calculated utility from an interaction, campsites will not only help increase the global happiness by contributing to positive interactions, but mainly prevent the biggest contributor of decrease from occurring.

Team-unique interaction heavily affect global happiness decrease

Unique interactions are a big contributor to the decrease of global happiness. These interactions often affect a large amount of guests, leading to the big effect these interactions have. While not all of these interactions directly affect happiness, like the fights started by AIK supporters, other interactions have indirect effects. An example

is the interaction by HIF supporters which brings more guests to a smaller amount of places.

Simulation if fairly balanced

From experiments we have made we can conclude that the various variables that are available to change in the simulation, all have significant effects. Changes to any of the main variables of the simulation will have visible effects on the global happiness of the simulation. Changes to one variable can still be balanced by changes to other variables, keeping the simulation in a satisfying mutualistic cycle.

To finally conclude the project as a whole we feel that the project has been a space for us to use and apply many of the things we have learned during earlier labs. It has also been a difficult, yet fun, challenge to work backwards as we have gotten a list of expected end results and having to come up with the idea of the simulation to meet these results.

It has been a satisfying way to tie the course together and we look forward to applying what we have learned in many future courses and situations to come.