Simulating a Festival with Q-Learning

Final Project in ID2209 Distributed Artificial Intelligence and Intelligent Agents

Hannes Rabo
School of Electrical Engineering and Computer Science
KTH - Royal Institute of Technology
hannes.rabo@gmail.com

Julius Celik
School of Electrical Engineering and Computer Science
KTH - Royal Institute of Technology
jcelik@kth.se

I. Introduction

This report presents a simulation made simulating a music festival using distributed artificial intelligence (AI) and multi-agent technology. The simulation was implemented in the modeling and simulation development environment GAMA. Additionally reinforcement learning was introduced to improve the decision making performance of the festival guest agents.

Five different kinds of movable agents with three personal traits were implemented to represent different festival guests with different behaviours and preferences. Two different kinds of locations were implemented which triggered different interaction behaviors between agents and their environment.

A. Agents

The five distinguished moving agents in the simulation were: normal people, party lovers, criminals, security guards and journalists. The specific behaviour is quite complex and many factors depend on each other. The primary factors that distinguish them from each other is listed in the table I.

The two locations in the simulation were bars as well as concerts which can be seen in figure 1. The bars are illustrated as white squares and the concert stages as black squares.

The two locations differed in how well agents felt while being there depending on their personal preferences, agent type as well as other factors. The primary factors bound to a specific locations are shown in table II.

B. Actions

At each time period, the agents were given a set of actions where they could perform any one of them with different outcomes. The actions contained things such as moving to another location, drink beer, drink water as well as dance on the current location. The actions are shown in table III and were the same for all types of agents. Difference in how they used them, is only visible after they learn to select what is best for them.

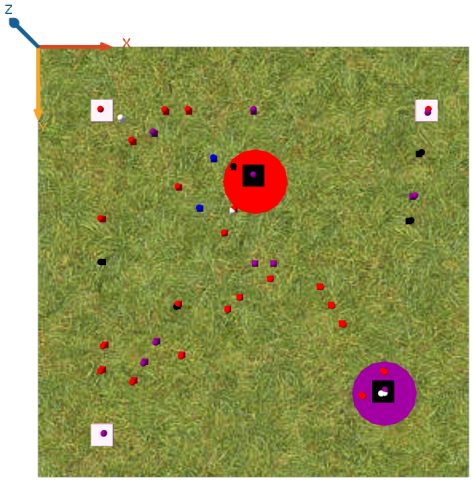

Fig. 1: Graphical representation of the map

II. Implementation

The simulation was implemented in GAMA with regular agents where each agent had a happiness value. The goal was to maximize the happiness for each agent using the reinforcement learning technique Q-learning and to create a scenario with the maximum number of happy agents. As each agent had the responsibility of optimizing the happiness of itself, it can not guarantee that a global optimum will be achieved, it is however an easy way of simplifying an otherwise complex situation.

The algorithm worked based on formula for Q-learning which updates a function $\mathcal{Q}(s, a)$ that approximates the choice of optimal action given a state s_t and a selected action a_t. Equation 1 shows how a new state is calculated from state, action and the parameters α (learning rate), γ (discount factor) as well as $\mathcal{R}_t$ which is the reward function.

$$\begin{aligned} \mathcal{Q}_{new}(s_t, a_t) \leftarrow & \\ & (1-\alpha) \cdot \mathcal{Q}(s_t, a_t) \\ & + \alpha \cdot (\mathcal{R}_t + \gamma \cdot max(\mathcal{Q}(s_{t+1}, a)) \end{aligned} \tag{1}$$

TABLE I: Agent behaviour

AGENT NAME	DISLIKES	LIKES
Normal	Party lovers when not drunk, Being close to criminals	Prefers bars over concerts, Being drunk
Party Lover		Prefers concerts over bars, Likes crowded places, Mildly likes to be close to criminals as criminals sell drugs
Criminal	Security guards	Being close to party lovers
Security Guard	Being drunk	Being close to criminals
Journalists	Being drunk	Being close to criminals, Music not in their taste

TABLE II: Location dependant parameters.

PARAMETER	COMMENT
place_closed	If the location has closed down due to fire or flooding (see section V about creative assignment)
crowded	If the place is crowded, fixed number of people required to hit this limit
has_security	If there is a security guard at the location
has_criminal	If there is a criminal at the location
has_partylover	If there are party-lovers closeby
music	The type of music that is playing

TABLE III: Agent actions

ACTION	OUTCOME
ACTION_GOTO_CONCERT_0	Set agent target to concert 0
ACTION_GOTO_CONCERT_1	Set agent target to concert 1
ACTION_GOTO_BAR_0	Set agent target to bar 0
ACTION_GOTO_BAR_1	Set agent target to bar 1
ACTION_GOTO_BAR_2	Set agent target to bar 2
ACTION_DINK_BEER	Get less thirsty and get more drunk
ACTION_DINK_WATER	Get less thirsty
ACTION_DANCE	Stay on location without any extra effect

For each time interval, equation 1 is used to update the state of the Q-table $\mathbf{Q}$ with the current state and action (s_t, a_t) as shown in equation 2.

$$\mathbf{Q} = \begin{bmatrix} \mathcal{Q}_{(1,1)} & \mathcal{Q}_{(1,2)} & \cdots & \mathcal{Q}_{(1,a)} \\ \vdots & \vdots & \ldots & \vdots \\ \mathcal{Q}_{(s,1)} & \mathcal{Q}_{(s,2)} & \cdots & \mathcal{Q}_{(s,a)} \end{bmatrix} \tag{2}$$

The effect from this is that the for each new state $\mathbf{s} \in \{1, \ldots, s\}$, we have an approximation of the reward of choosing action $\mathbf{a} \in \{1, ..., a\}$. From this approximation function $\mathcal{Q}(s, a)$ we can choose the action in our current state row of $\mathbf{Q}$ that has the highest value and thus the highest likelihood of producing a state that gives the highest reward.

For our example, the reward $\mathcal{R}$ was calculated after each step in the simulation based on agent preferences, close agents and environment as discussed in the introduction.

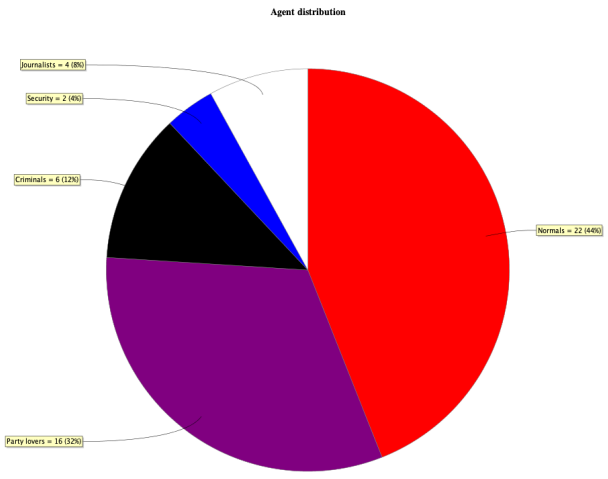

Fig. 2: Distribution of agents

III. Experiments and Results

To be able to compare results between changes in the simulation we used a constant randomness seed. The agent distribution is shown in figure 2.

We traced happiness levels of different groups of agents, general mean happiness of all of the agents, drunkenness levels, as well as concert and bar attendance. Happiness was used as

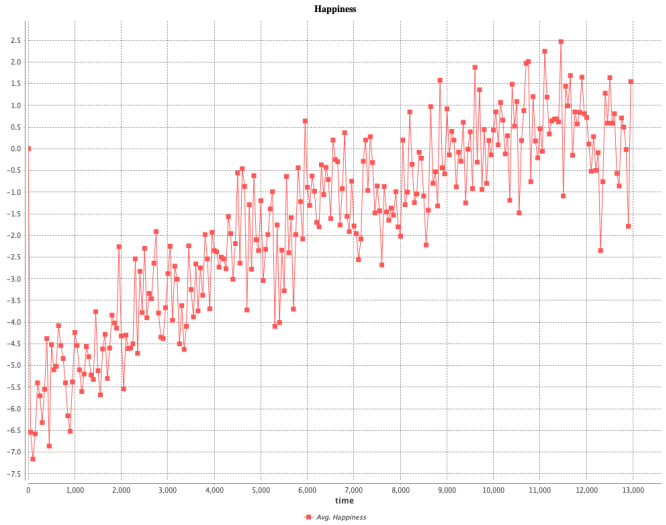

Fig. 3: Average happiness of all agents of the simulation without training

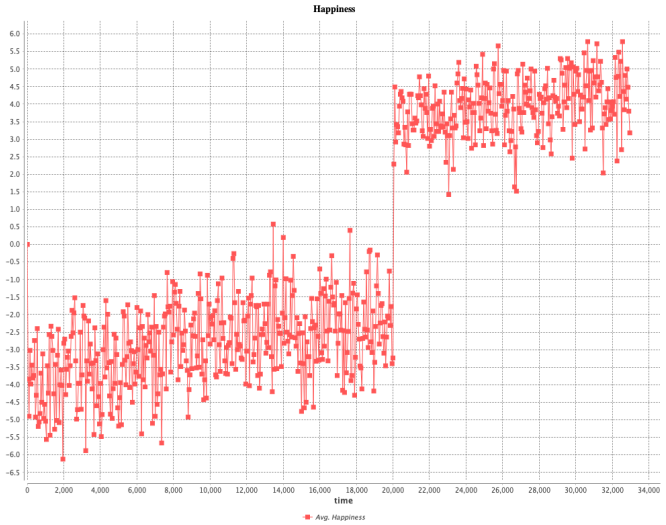

Fig. 4: Average happiness of all agents of the simulation with training

a goal metric to optimize and the other metrics were used to better understand the behaviours of the agents. Drunkenness levels tells us how good the agents were at maintaining a healthy alcohol consumption, and concert and bar attendance shows us if the agents learns that closed locations are bad places to go to.

The data that we had showed us that the agents happiness improved over time, and we theorized that their happiness would keep improving even more if they had more time to learn. This is because we saw no indication of the happiness levels converging towards a peak happiness limit. Therefore we added a training phase where the agents acted randomly to try all kinds of states to therefore rate different states before the simulation.

A. *Happiness*

The mean happiness of all agents are shown in the figures 3 and 4. While figure 3 shows that some increase in happiness happens over time when using the Q-table to choose action from start, this limits the number of states the agents will try. As soon as they start to use actions in a state, there is a high probability that the state gets a higher value when using the function $\mathcal{Q}(s, a)$ which means that random exploration of new actions are slow. If we instead let the agents roam around freely for a number of steps, we ensure that they have explored sufficiently to have a clear advantage while calculating the best actions (figure 4). In the figure we can see that the happiness is hovering around 0.5 for agents without training while it is close to 5 after a training episode. Due to this behaviour, all results are presented with a training phase up to time period 20'000 and regular simulation after.

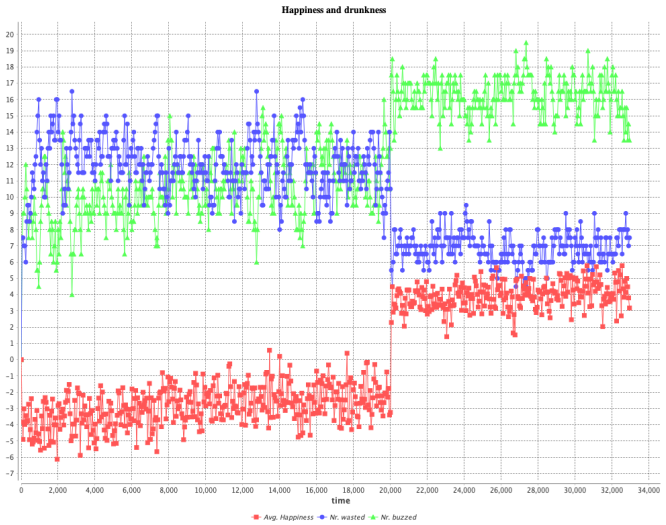

Fig. 5: Drunkenness of agents and happiness. Drunkenness is displayed as a count out of 50 total where the happiness value is the mean value for all agents in the range [-10, 10].

B. *Drunkenness and Happiness*

While observing the drunkenness level of the agents, we can see why more happy people appear as already shown in figure 4. Figure 5 shows how agents learn to not get "wasted" (low reward) but most gets "buzzed" (high reward) which has a direct effect on the mean happiness. Other factors are also prevalent, this is however one of the major reasons.

IV. Discussions and Conclusion

In every measurable metric we can see that adding a training phase greatly increased performance for all agents. If the festival was active for a longer time, we would probably be

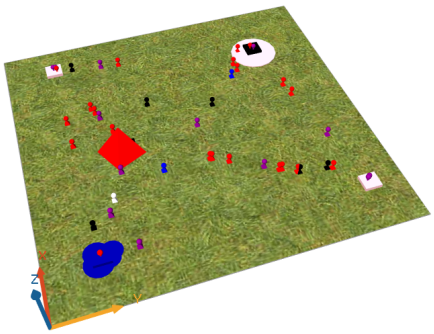

Fig. 6: Graphical representation of the simulation with flooding bar as well as burning stage

able to get closely similar performance. We also used constant α (learning rate) and γ (discount factor) parameters in our Q-function (Equation 1), instead of variable. If we started with a low learning rate and increased it during execution, agents would learn more from their experiences in the beginning, whilst being better at retaining knowledge later on. This could be another method for increasing the performance of the agents during the simulation.

V. Creative implementations

For the creative implementations we made it possible for bars to flood and stages to burn. That is two different creative tasks. Figure 6 shows a flooding bar (blue, lower left) as well as a burning stage (red, more central). This had the effect that all types of services provided at the locations were turned of and rewards removed if agents entered these locations. The goal was to see if this effected where agents were spending their time.

The trends for bars is shown in figure 7 which shows that most normal agents choose to be in a bar when training stops. They will however avoid the bar that is flooded. As some randomness is added to how agents move to improve the results and encourage further learning, we can see that there is always a small number of agents present at the location, however they leave the next tick as they realize that the place is closed.

The same type of trends can be seen with the ”party lover”-agents that are more inclined to be at concerts than normal agents. Here we see that most ”party lovers” choose to go to a concert after training, while they avoid the burning stage.

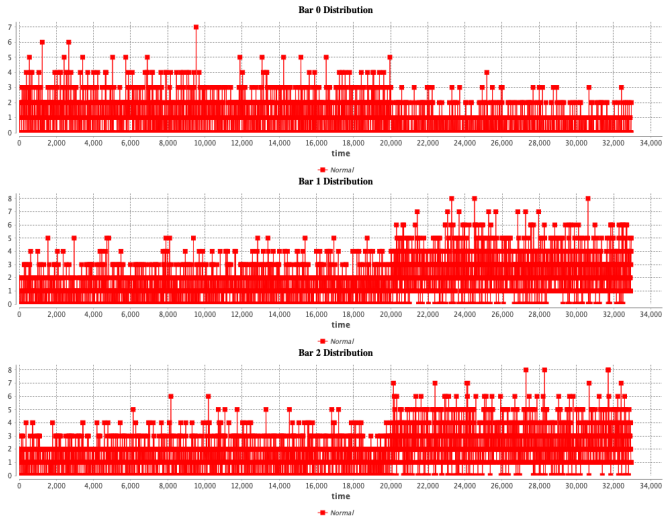

Fig. 7: Number of ”normal”-agents present at the different bars with 20’000 ticks of training. Bar 0 is flooded in this example

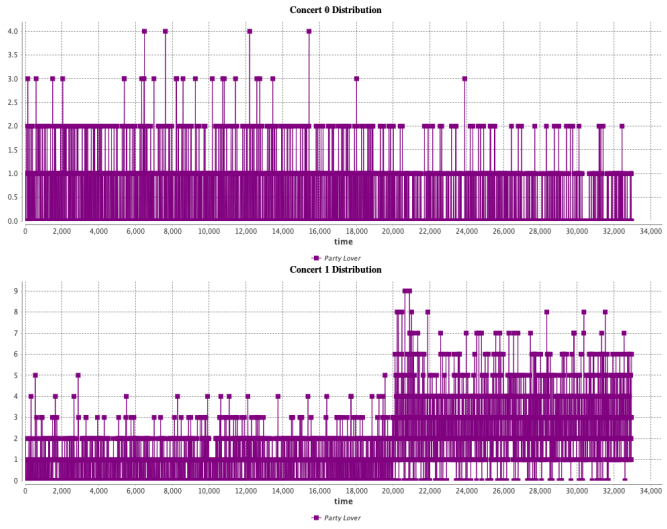

Fig. 8: Number of ”party lover”-agents present at the different bars with 20’000 ticks of training. Concert 0 has burnt down in this example