Multi-Agent Systems: Final Project
Mafia game simulation

Minh Ngo
10897402
University of Amsterdam
minh.ngole@student.uva.nl

Georgi Terziev
11128739
University of Amsterdam
g.d.terziev@gmail.com

Arthur Bražinskas
11138904
University of Amsterdam
arthuras.brazinskas@student.uva.nl

March 27, 2016

Abstract

In this project the party game Mafia is simulated. The simulation considers a fraction of mafias to innocents and different types of personalities (i.e. naive, vengeful, and logician) to research the influence of those parameters on the game. The simulation is implemented in the belief-desire-intention framework with probabilistic base for decision making, such as a voting heuristic.

1 Introduction

Mafia is a popular party game that models a conflict between an informed minority, the mafia, and an uninformed majority, the citizens (innocents) (Wikipedia). At the start of the game, each player is secretly assigned a role affiliated with one of these teams. The game has two alternating phases: night, during which the mafia may covertly "murder" an innocent, and day, in which surviving players debate the identities of the mafiosi and vote to eliminate a suspect. Play continues until all of the mafia members or all of the citizens have been eliminated. And in those cases the game is won by citizens or mafias respectively. Depending on a decision making strategy ("personality") of each player and a fraction of mafias to innocents different outcomes may happen. The research question about influences of those parameters will be reflected in the following report.

To be able to draw conclusions, we built a Belief-Desires-Intentions (BDI) based multi-agent system simulator for the game. In every simulation there are multiple agents, each corresponding to a single player. All the citizens agents have a common goal to eliminate all of the mafia agents. They need to collaborate and communicate in order to do so. Conversely, the mafia agents need to communicate and collaborate in order to achieve their goal, which is to eliminate all the citizen agents. In order to do so, they are also allowed to lie to the citizen agents. The agents act on the environment by eliminating one agent per game phase. In order to choose the agent to be eliminated, they vote simultaneously and their votes are counted in a plurality vote. In order to choose who to cast its vote for, each agent has its own beliefs. As explained in section 2.4, agents with different personalities update their beliefs in a different manner. Moreover, as explained in section 2.5, they use their beliefs differently to choose who to cast their vote for.

An important special action of the agents is communicating with each other and sharing their suspicions. Right before agents cast their votes, each of them points to somebody they suspect as being a mafia agent and share the id of that agent with everybody else. Agents with different personality types incorporate this newly acquired knowledge differently, however they have to be careful who they believe, since the mafia agents will try to deceive them, as explained in 2.3.

2 Background

2.1 Architecture and BDI framework

The game can be understood as a loop of steps presented in a list below. The game starts at the night stage and the mafia members kill their first victim.

- Night Phase
  1. Mafia pretend to go to sleep and citizens go to sleep
  2. Mafia votes
  3. The player who gets most votes is eliminated
- Day Phase
  1. Citizens wake up and mafia pretends to wake up
  2. Opinions (suspicions) exchange
  3. All players vote
  4. The player who gets most votes is eliminated

Belief of players about other players in our project can be of 2 types: who is mafia and who is dangerous for a particular player. An additional belief about what phase of the simulation we are currently in is updated based on a perception from the environment. Depending on that belief each player's desires are updated to either "find mafia" and "sleep" for citizens, or "hide" and "kill citizens" for mafia. Based on desires agents update their intentions that may be "vote", "awake", "exchange opinion", "kill", "eliminate", "pretend to sleep", and "pretend to wake up" depending on the agent's role.

The game continues until the termination criteria is met, for example, all mafia members are eliminated or winner is obvious (e.g. 1 citizen and 2 mafias left).

The beliefs are updated in the beginning of multiple steps. First we update beliefs in step 1 of the night phase because that's when mafia has to consider the outcome of the day phase in order to cast their votes for the night's kill. Second, we update beliefs in step 1 of the day phase, when everyone wakes up (or pretends to wake up) and beliefs should change after the previous day's vote and the night's kill. Finally, we update beliefs a third time in step 3 of the day phase, as after communication some valuable hints are presented and have to be incorporated into the player beliefs.

2.2 Personalities

Since the mafia game has a deep psychological foundation where people use their intuitive judgments to uncover mafia members or lie to hide their mafiosi identities, the simulation reflects that by assigning different personality types for each player. The first type of players is naive, those players trust everyone and are highly affected by social influence. For example, a naive citizen will incorporate other players' suspicions as golden truth. The second type is vengeful, those players get offended if someone votes against them and are willing to avenge instead of rationally consider facts. Therefore, the player of vengeful type might choose to vote against a player even if the likelihood of him being a mafia is very low because that player voted against him several times before. In addition, the vengeful player forgets offends slowly, but is less affected by social influence, and considers voting of other players as clues. For example, if a player votes against a citizen that could be a clue that the player is a mafia agent. Finally, the last type is logician, those players are not offended as much as vengeful, and consider previous voting of other players as clues. As previously mentioned, the personalities are encoded in the way beliefs are updated and voting casting heuristics which are described in sec. 2.4 and sec. 2.5 respectively.

2.3 Communication

The communication phase when players exchange their suspicions (opinions) about who they think is mafia can be a valuable source of information for citizens. The communication algorithm is implemented as a result sharing algorithm where agents are sharing their local view to achieve a better overall global view Wooldridge (2009). The local view is a product of derivation made by each agents but not all their internal states. In addition each agents have different goals in the game and therefore will try to share either real local view or the fake local view and an agent-receiver should decide either to consider these results or not dependent on the ”reliability” of the sender.

If a certain player accepts another player’s opinion during communication, that will mean that it will be considered during voting by updating social beliefs. Citizens express their opinions by naming players who have highest probabilities to be mafia according to their beliefs. However, mafia is well informed about who are citizens and has no interest in exposing their mafia team-mates. Therefore, we propose a lying schema that encodes mafias willingness to confuse citizens by presenting opinions that are far from truth. Concretely, mafia players name citizens who are most dangerous based on their beliefs in order to provoke other citizens to vote against their own team-mates who could correctly suspect who are mafias. In addition, the mafia players always reject citizen opinions, and only consider opinions of mafia team-mates. On the other hand, it is irrational for citizens to believe players who are very likely to be mafia. Therefore, we encode an idea that each citizen believes other players proportionally to the complement of their danger factor. Mathematically, it can be expressed as follows, let Y_i be a random variable corresponding to a player i believing another player, For example, event $Y_i = k$ means that the player i beliefs the player k, and will therefore take his opinion into consideration during voting. We can simulate whether a player should believe another player by performing Bernoulli trials as shown in eq. 1, where D_k corresponds to the danger factor of a player k. It is worth noting that danger factors (beliefs) are updated with consideration of personalities, for example naive players believe everyone and therefore accept all opinions.

$$P(Y_i = k) = (1 - D_k) \tag{1}$$

2.4 Beliefs update

Belief of each agent about another agent’s role is updated based either on the environment observation (previous actions performed by agents) or social influences (information obtained from other agents). In the second case information can have noise (produced for instance by mafia to sabotage the society or by wrong conclusion drawn by other citizens from their observations) that requires consideration from the agent’s side.

2.4.1 Social influence

The social influence beliefs update is directly based on the communication phase which is described in section 2.3. Once all accepted opinions are calculated, histograms are created with frequencies of votes against each player, and normalized to sum to one. Intuitively, if we compute the normalized histogram for player 3, and there are 2 opinions against player 1 and 3 opinions against player 2, then the normalized histogram will be (0.4,0.6,0)[1].

2.4.2 Danger updates

Each agent has beliefs about which agents are dangerous for him. For mafia those will be agents that voted against them in previous rounds, for a particular citizen dangerous are considered people that voted against him. We consider the following heuristic for danger belief update:

$$danger_i = danger_i + k * danger_i(1 - danger_i) \tag{2}$$

where $danger_i$ is the probability of the i-th agent to be dangerous for a particular agent, k is 0.2 for vengeful agents and 0.1 in the other cases (vengeful personalities are more spitfire). The update in this formula is very sensitive to values around 0.5, making it difficult for agents to be ambivalent to who is dangerous for them.

In addition, the probability will degrade steadily, following the formula:

$$danger_i = k * danger_i \tag{3}$$

where k is 0.9 for vengeful agents and 0.8 otherwise (vengeful personalities remember "harms" longer).

2.4.3 Mafia belief updates

Belief about who is mafia is updated in the same way as described in formulas 2, 3, but the difference is that an agent updates his belief dependent on previous votes of other agents in general. If the agent A has voted against the agent B in the previous round and it turns out that B is a citizen, but not a mafia, then A will have a higher chance to be a mafia. In addition, agents that are mafia don't need to update their belief about mafia since they already know their partners.

2.5 Voting

In order to perform voting, we use an *argmax* social choice function. Each player has associated probability preferences towards eliminating (voting against) other players. The probability that can be interpreted as willingness of the player to vote against other players and the function itself is a heuristic that we describe in this subsection. Let X_i be a random variable corresponding to the choice of player i to vote against another player, for example $X_i = k$ is the event where player i decides to eliminate player k. We are interested in returning the id of the player that player i wants to eliminate the most, which is achievable using *argmax* operation over the probabilities associated to every player that player i wants to eliminate, as shown in eq 4. Finally, we can compute the full probability using the law of total probability by conditioning on 3 types of beliefs (M: mafia, D: danger, S: social influence) as shown in eq. 5, or the simplified version in eq. 6.

[1] We also ignore opinions against the current player, because it makes no sense to be socially influenced to vote against yourself.

$$k^{'} = \arg\max_{k} P(X_i = k) \tag{4}$$

$$P(X_i = k) = P(M_i)P(X_i = k|M_i) + P(D_i)P(X_i = k|D_i) + P(S_i)P(X_i = k|S_i) \tag{5}$$

Intuitively, the voting heuristic allows us to encode both personalities and how beliefs influence the decision making (voting) by breaking down the full probability into different conditional and prior parts. Concretely, for the prior probabilities (e.g. $P(S_i)$) we shall encode personal aspects, such as how much player i is influenced by social factors. For the conditional probabilities we shall consider objective beliefs that player i has. For example, $P(X_i = k|D_i)$ tells us how likely it is that player i thinks that player k should be eliminated considering beliefs about danger factors.

$$\lambda_i^1 P(X_i = k|M_i) + \lambda_i^2 P(X_i = k|D_i) + \lambda_i^3 P(X_i = k|S_i) \tag{6}$$

The previously mentioned probabilities are calculated in the following way: the priors are empirically chosen constants for each personality, for example, naive player has $\lambda_i^1 = 0$, $\lambda_i^2 = 0.5$, and $\lambda_i^3 = 0.5$, and conditional probabilities are normalized beliefs, e.g. probability of voting against a player given mafia beliefs is directly based on how likely it is that the player is mafia. Since player beliefs change over time, the heuristic will cast different probabilities depending on the beliefs.

3 Experiments

Two experiments have been conducted. In the first one personalities are assigned randomly. For each configuration about 500 games have been run after which a probability of mafia winning the game is computed. We considered amount of mafia from 1 to 4 and amount of citizens in the range from 3 to 10 and from 19 to 21. A game is considered balanced when the ratio between mafia and citizen agents allows each team to win the game is considered as a game that has a probability of mafia winning the game around 50% of the times.

The second experiment has considered different player personalities. We have picked the standard Mafia game configuration (2 mafia vs 10 innocents). And run experiments for 9 combinations where we assigned mafias or citizens into one of personality categories "naive", "vengeful" or "logician".

4 Results

The results from our experiments are reported in tables 1 and 2. A quadratic dependency between the amount of citizens against mafias can be seen in [Table 1]. Other configurations make the game less stable and give one of sides a higher chance to win. With an increased amount of citizens the mafia starts to loose their domination.

In the second experiment an influence of personalities can be seen. Logician and vengeful players dominate naive players. It can be noticed that with logician citizens mafia has significantly less chance to win. With naive innocents the chance of winning is about 20% higher. There are some clues of unbalanced game that we can see in the second table, that we specifically address in the next section.

citizen count \mafia count	1	2	3	4
3	**0.55**	0.85	-	-
4	0.39	0.86	0.97	1
5	0.38	0.71	0.95	1
6	0.23	0.72	0.93	1
7	0.31	0.59	0.93	0.97
8	0.21	0.58	0.8	0.97
9	0.34	**0.52**	0.81	0.96
10	0.17	0.45	0.75	0.87
19	n/a	n/a	**0.5**	0.83
20	n/a	n/a	0.57	0.75
21	n/a	n/a	0.68	0.76

Table 1: Probability of the game to be won by mafia for different amount of players. Best game configurations are marked in bold.

mafias \citizens	**naive**	**vengeful**	**logician**
naive	0.47	0.41	0.28
vengeful	0.51	0.47	0.37
logician	0.52	0.46	0.36

Table 2: Probability of the game to be won by mafia for different types of personalities.

5 Discussion

After performing a number of experiments we discovered that the heuristic and the current set of beliefs could give a bigger advantage to citizens than to mafia. For example, table 2 shows that the minimum probability of mafia to win is 0.28 and the maximum is 0.52, while for citizens they are 0.48 and 0.72, which can be an indicator for the game being unbalanced. The game balancing is non-trivial task, especially when probabilistic decision making heuristics are involved. We hypothesize that personalities give a bigger advantage for uninformed players, for example that they are capable of inferring who is a trustworthy source of information, and who is potentially mafia, while mafia by being informed looses those benefits. It's also possible that the corner case scenarios give an advantage to citizens on average. Finally, it's also possible that 10 citizens and 2 mafia players in previously mentioned personality brackets give an advantage to citizens. All those questions will be considered in the future work.

6 Conclusion

In the project we have presented and implemented the mafia game's simulation in the belief-desire-intention framework and introduced several probabilistic approaches that can be used to encode psychological aspects of the game, such as the heuristic voting function, communication phase, and beliefs updates. In addition, we have presented empirical results that shed light on how personalities and mafia to citizen proportions influence the game outcomes.

A fraction of citizens in the game that is a squared amount of mafia turns to be a good configuration where both mafias and citizens have an equal chance to win. Personalities of players have a huge impact into the game outcome. A team tends to lose more frequently with naive players as members. Logicians in the team increase a chance of winning. However, the Nash equilibrium cannot be easily found for a game of such complexity.

7 Future work

7.1 Non-simultaneous voting

In the project we performed a simultaneous plurality voting that can be a naive approach in comparison to the real game, and in the future work it would be interesting to model how player beliefs change once they obtain new information about votes of other players before casting their own vote.

7.2 Priors estimation

In sec. 2.5 we introduced a voting heuristic that relied on prior knowledge specific to each player type, and it is feasible to perform a machine learning parameters optimization in order to fit a best set of parameters for each strategy. Intuitively it can be either done by extracting log of correct and incorrect cotes and defining an objective function, and finally taking a partial derivative of the objective function w.r.t. the parameters of interest and updating those parameters using gradient descent or stochastic gradient descent (Bottou, 2010).

7.3 Different roles

We did not consider other roles except for mafia and citizens in our experiments. Including other roles, in addition to personality types, will create several more dimensions in the search space, and therefore will be left as future work.

References

L. Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*, pages 177–186. Springer, 2010.

Wikipedia. Mafia (party game). URL `https://en.wikipedia.org/wiki/Mafia_(party_game)`.

M. Wooldridge. *An Introduction to MultiAgent Systems*. Wiley Publishing, 2nd edition, 2009. ISBN 0470519460, 9780470519462.