Multi-agent System in Fire Emergencies

Project Report - Group 53

Joana Raposo
Instituto Superior Técnico
Lisboa, Portugal
joana.raposo@tecnico.ulisboa.pt

João Marto
Instituto Superior Técnico
Lisboa, Portugal
joaopomarto@tecnico.ulisboa.pt

Larissa Tomaz
Instituto Superior Técnico
Lisboa, Portugal
larissa.tomaz@tecnico.ulisboa.pt

ABSTRACT

Nowadays it is normal to evaluate the performance of artificial intelligence agents in tasks that are conventionally performed by humans, either with the objective of optimizing the performance of human agents or replacing humans with robots in these tasks.

Fires are a catastrophe that humanity has been dealing with for many years and most of the time it ends in tragedies with countless victims and environmental and financial damage. This project aims to study the behavior of agents when fighting the fires and to optimize it through cooperation.

KEYWORDS

Fire; Multi-Agents; Decision Making; Artificial Intelligence

1 INTRODUCTION

This project was developed in the context of the Autonomous and Multi-Agents Systems course at Instituto Superior Técnico. Our main objective was to study different types of strategies and behaviors of agents in a firefighting situation. As it is a multi-agent system, the basis of our experiment is the collaboration between the agents.

Social ability is the ability to interact with others. Such ability is commonly obtained in multi-agent systems through cooperation, coordination and negotiation between agents. In our system we can identify those three types of interaction. Cooperation as all agents work together to achieve a common goal: to put out all the fires. Coordination because agents have to coordinate their actions in order to make fire fighting as efficient as possible. And finally, negotiation because the agents must come to an agreement, for example: which agent has priority over choosing which fire to fight.

1.1 Definition of the problem, requirements and objectives

Our project models two agents in an environment with multiple fires, as it can be seen in Figure 1. The agents must cooperate in order to fight all fires in the shortest time-steps possible. Their task is completed once the fires are all fought or the maximum time-step established is achieved.

AASMA Project, IST, 2022, Lisbon, Portugal

https://doi.org/

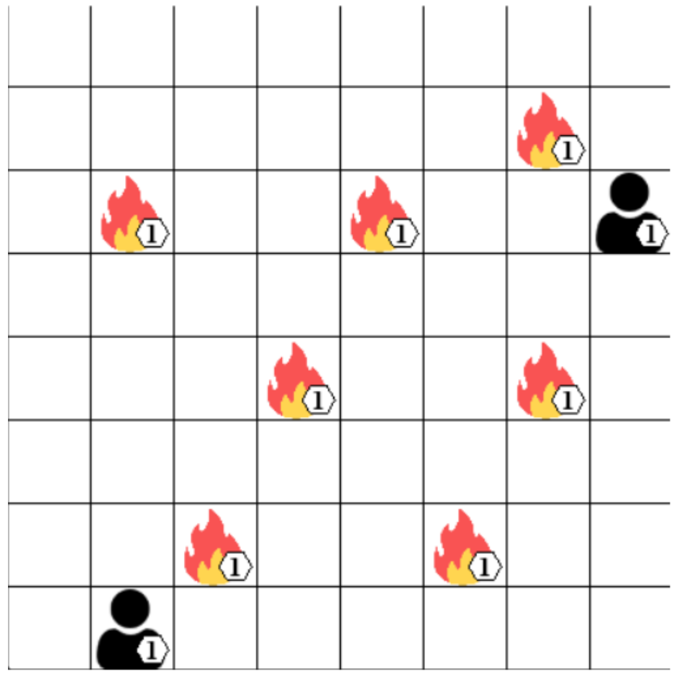

Figure 1: Environment of our project: 8x8 grid with two agents and several fires

2 CONCEPTUAL MODEL OF THE MULTI-AGENT SYSTEM

2.1 Agent and Environment properties

2.1.1 Sensors and Actuators. Each agent has sensors that allow them to see the position of existing fires and the position of all other agents.

Regarding actuators, agents are able to stay in a fixed position or move around the environment and have a mechanism that enables them to fight a fire when standing next to it.

2.1.2 Environment Details. The environment starts with seven fires, in random positions of the grid. The initial positions of the agents are also random.

What makes the environment more dynamic is that over time new fires appear in random positions on the map and agents have to include these new fires in their decisions.

2.1.3 State Space. The state space is the combination of all possible options of

(fx1, fy1, ... , fxN, fyN, ax1, ay1, ax2, ay2), where:

- N is the maximum number of fires that can be generated
- *fx1, ..., fxn* are the row coordinates of the fires (if less than N fires are currently showing, then the position is -1)
- *fy1, ... , fyn* are the column coordinates of the fires (if less than N fires are currently showing, then the position is -1)
- *ax1 and ay1* are respectively the row and column coordinates of the agent 1
- *ax2 and ay2* are respectively the row and column coordinates of the agent 2

2.1.4 Action Space. The action space is composed by a total of 6 actions:

- **None** - The agent remains in the same cell
- **North** - The agent moves up
- **South** - The agent moves down
- **West** - The agent moves to the left
- **East** - The agent moves to the right
- **Fight** - If a fire exists in a cell adjacent to its current position, the agent fights the fire and it disappears from the environment

2.1.5 Observation Space. The observation space is equal to the state space: both agents have full observability, knowing the positions of all fires and the positions of all agents.

2.1.6 Decision Making Behavior. In our project we developed four different decision making behaviours:

- Random
- Greedy
- Greedy with coordination
- Greedy with social convention

This different behaviors will be discussed with more detail further in section 3.

2.2 Architecture of the Agent System

The agents have a deliberative architecture, since each agent main task is to stop the fire, if one agent is already fighting the fire, the other agent should not interfere.

Regarding commitment, it can be considered that the agents architecture follow a single-minded commitment, since they either maintain their intention of fighting the fires until they achieved it or until it is no longer possible to achieve it (when the limit of time steps are achieved).

3 TYPES OF AGENTS BEHAVIORS

3.1 Random Agent

As a baseline, we first designed our agents to choose an action at each time step randomly.

In this type of behavior, the agent does not make use the fire coordinates of the observation space. It only chooses random actions, that can be successful or not (for example, if it chooses the action FIGHT and there is no adjacent fire, the action has no impact. Or if it tries to move to a cell where there is currently a fire or another agent, the action also fails and the agent stays in the same cell).

For this type of agent to succeed in its task, it must be positioned in a cell adjacent to a fire and make the correct decision to fight it.

3.2 Greedy Agent

This approach is our first improvement of the baseline agent type.

At each time step, both agents calculate their Manhattan distance to all of the fires available. After this, they both choose the fire with the smallest distance and choose the action that will make the agent move towards the fire. When the Manhattan distance returned to the closest fire indicates that the agent is in one of the adjacent cells, the agent chooses the action FIGHT.

3.3 Greedy with Coordination Agent

As an improvement of the greedy version, we focused on solving the problem of both agents aiming towards the same fire.

To do this, we made our agents work in different locations by restricting those agents to a certain zone, in the case of a execution with two agents, each agent gets half the grid. By doing this we ensure that there are no conflicts in path and both agents fight their fires alone, meaning there won't be two agents fighting the same fire or trying to go to the same cell, therefore making them fight fires more efficiently.

The base behavior of the agents continues to be greedy, that is, each agent tries to fight the fire closest to itself, but now within its spatial restrictions.

3.4 Greedy with Social Convention

As another improvement to the greedy version, and again with the aim of maximizing the agents efficiency by not going towards the same fire, we decided to use a **social convention**.

In this way, at the beginning of the execution of every episode, an agent order is defined (randomly). The agent that comes first in this order can be seen as the leader, that can choose first which fire it goes to.

As the Greedy version, the leader calculates a Manhattan distance to all the fires and chooses the nearest fire to move towards to. After that, it is time for the second agent to decide which fire to pick. If the nearest fire is the same as the leader, he must recalculate and choose the second nearest fire. If there are no fires left for the second agent, he picks randomly its next action.

4 COMPARATIVE ANALYSIS

To evaluate our system, we compared between the different types of agent behaviors with two different metrics:

- Average number of **steps** per episode until all fires are fought.
- In a time range fixed for ten steps, **how many fires** the agents can fight.

The first metric allow us to understand how quickly agents can achieve their main goal: **fight all the fires** and leave the environment out of danger.

The second metric enhances the capability of the agents to **find the fires** and fight them.

4.1 Average steps per episode

For this evaluation, we set a limit of fifty steps per episode, and run the model for fifty episodes in each agent mode.

4.1.1 Expectations. We expected that a Random Agent would not be able to finish, since the number of fires is much greater than the number of agents. With these type of agents not making use of the observation space when making their decision, it is really hard to complete their goal - **fighting all the fires**.

Regarding the Greedy Agents, we expected that they would be able to finish this task, since they move towards the fires. However, we predicted that there would be some moments that both agents would move towards the same fire, causing inefficiency.

Having that in mind we developed two different coordinated Greedy Agents, as detailed before in sections 3.3 and 3.4, with the aim of solving this issue that turned out to be very recurrent. We expected that this versions would be better than the greedy version, but very similar between themselves.

4.1.2 Results. The obtained results are shown in Figure 2.

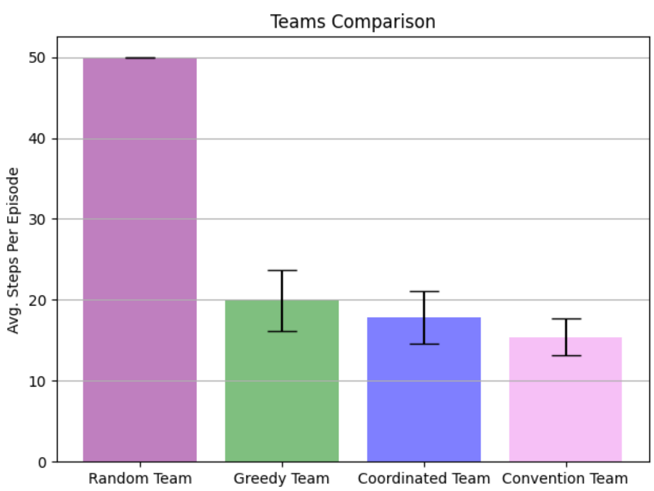

Figure 2: Average steps per episode results of the different agent types

The average steps per episode obtained were:

- **Random Team** - 50.00 steps
- **Greedy Team** - 19.88 steps
- **Coordinated Team** - 17.84 steps
- **Convention Team** - 15.42 steps

And their respective standard deviation:

- **Random Team**: 0
- **Greedy Team**: 13.55
- **Coordinated Team**: 11.50
- **Convention Team**: 8.04

As expected, Random Team was not able to finish in 50 steps, which leads to a standard deviation of 0 (all agents reach the 50 steps limit).

Also as anticipated, the Greedy Team was a positive improvement when compared to the Random Team. It has a bigger standard deviation, since the randomization of the problem makes it very different from one episode to another: In some episodes, it is more likely to happen situations where the agents are well dispersed and always aim to different fires, in others they may move almost together and aim for the same fire.

What we were not expecting was the Convention Team to actually have better results than the Coordinated Team. After analysing the executions, we understood that in the Coordinated Team (3.3) sometimes the distribution of the fires was not even, leading to one agent being responsible to fight much more fires than the other one. In addition, the spatial restriction makes an agent not able to put out a fire that is adjacent to it if this adjacent cell belongs to the space destined for the other agent, which ends up making this behavior less efficient. In the Convention Team (3.4), the agents are always assigned to a different fire (that is always the closest possible), leading to less unnecessary/inefficient movements and to less moments of an agent without a fire assigned.

Once again, the standard variance of the Coordinated and Convention Teams are mostly due to the randomness of the positions of the fires, agents and randomness in the appearance of new fires. Besides that, it is lower than in the Greedy Team, suggesting that these agent versions are more stable and find less conflicts.

4.2 Average of fires fought in 10 steps per episode

In this evaluation, we set a limit of ten steps per episode, we kept the number of fires fixed (did not generated new fires besides the initial ones) and run fifty episodes.

4.2.1 Expectations. We expected that a Random Agent would be the agent type that would fight less fires.

After observing the results from section 4.1.2, we also anticipated that the Greedy agents would be able to perform better, and that both Coordinated Teams (3.3, 3.4), could fight even more fires than the Greedy Team.

4.2.2 Results. The obtained results are shown in Figure 3.

The average of fires fought per episode obtained were:

- **Random Team** - 0.88 fires
- **Greedy Team** - 5.14 fires
- **Coordinated Team** - 4.72 fires
- **Convention Team** - 5.28 fires

And their respective standard deviation:

- **Random Team**: 0.74
- **Greedy Team**: 0.87
- **Coordinated Team**: 1.50
- **Convention Team**: 1.34

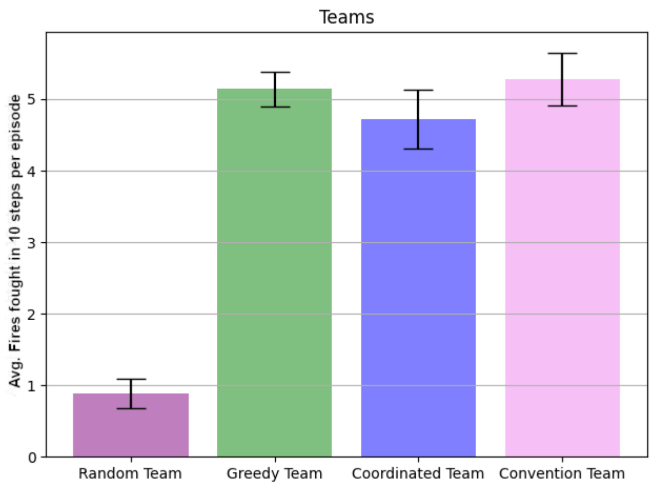

Figure 3: Average fires fought in 10 steps per episode results of the different agent types

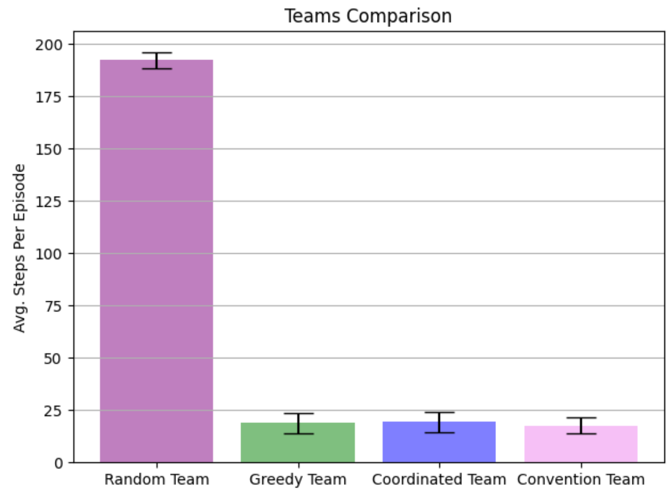

Figure 4: Execution with a limit of 200 steps

As we predicted, the Random Team shows a big difference when compared to the other teams.

The Greedy Team results show a big improvement to find the fires and fight them, since this team actually moves in the direction of the nearest fire.

The Convention Team showed to be slightly faster to than the Greedy Team, and this can be explained with the fact that since the two agents always aim for different fires, they can reach more fires.

What was not expected was the lower results by Coordinated Team. After watching the executions, we understood that the reason for this is a consequence of the fact that sometimes even if a fire is next to a agent, the agent might not fight that fire because it does not belong to its half of the environment. In addition, sometimes the agents were positioned in the other half of the environment that they were assigned to. Therefore, their first fires sometimes were very far, causing the agents to take more time to reach the fires.

4.3 Extra Experiments

We also executed some additional tests, where we varied the number of agents and the number of steps per episode.

The first thing we could observe by increasing the number of steps per episode incrementally, was that the random agent takes about 190 steps to finish, as we can see in Figure 4.

One interesting experiment was to compare the execution of greedy and random agents when varying the number of agents collaborating. From this experiment we verified that with the collaboration of teams with seven agents the random team caught up with the greedy team, as can be seen in Figure 5.

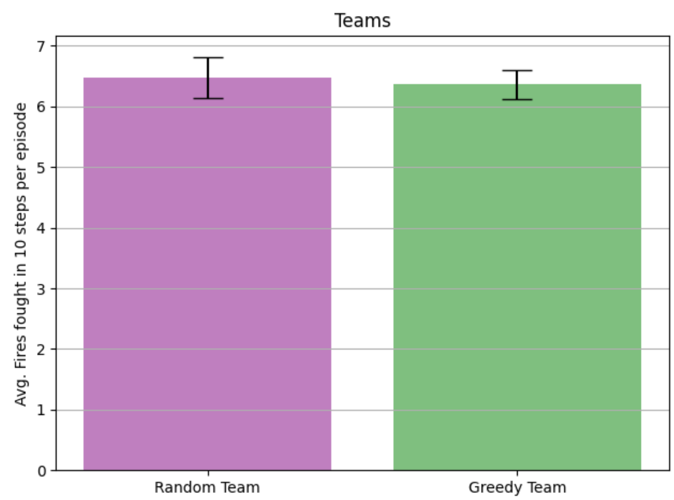

Figure 5: Seven agent experiment

The average of fires fought per episode obtained were:

- **Random Team** - 6.48 fires
- **Greedy Team** - 6.36 fires

And their respective standard deviation:

- **Random Team**: 1.22
- **Greedy Team**: 0.89

We believe that this results happens because when the number of greedy agents increase, they tend to focus on the same fires and end up being less effective, in addition to the usual conflicts of actions (ex: agents who try to move to the same cell to fight a fire).

5 FUTURE WORK

In this section we present several improvements that could be made to further improve our research and also make better contributions to different situations of real world scenarios. These were not implemented due to time constraints.

The next step in our research would be to extend the social conventions algorithm to any number of agents. In addition, we also want to test implementations with communication mechanisms between agents.

We would also want to take the research to the next level by introducing level two fires. In other words, fires that would require the collaboration of two agents for their extinction.

We already have a basic implementation with both random and greedy agents, as it can be seen in Figure 6.

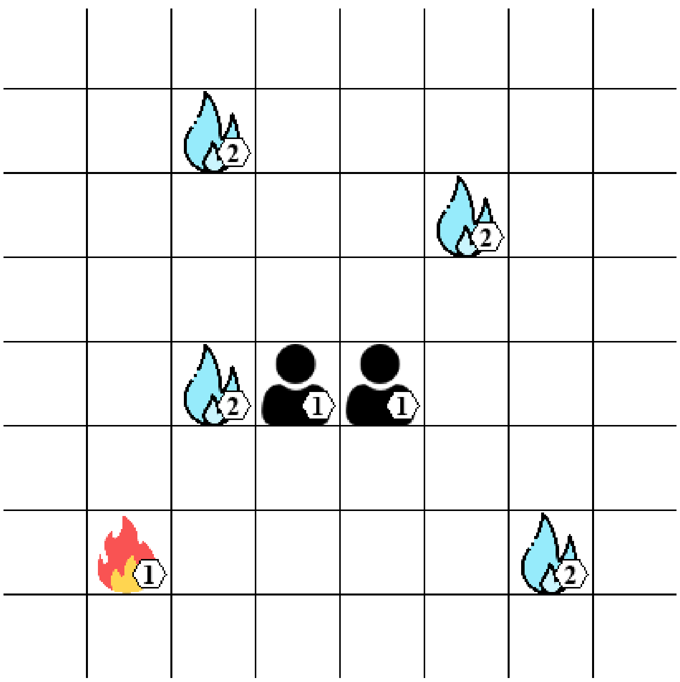

Figure 6: Implementation with fires of level two

In this scenario, the random and greedy implementations become quite ineffective because in most of the executions the agents are stuck in level two fires waiting for the other agent to collaborate, as can be seen in Figure 7. Such behavior will have to be avoided through the implementation of new coordination and communication mechanisms.

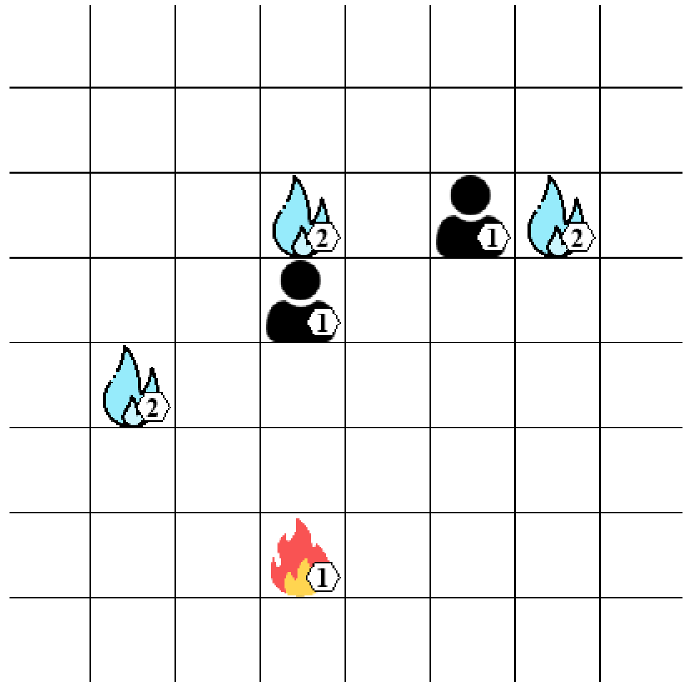

Figure 7: Agents stuck to different fires of level two

6 CONCLUSION

From the comparative analysis results (section 4) we can conclude that the type of agents that showed the best performances were the **social convention agents**, since they have proven to achieve the goal of fighting all the fires faster and also to find and fight more fires in a given amount of time.

We can also infer with the obtained outcomes that our solution was adequate to the given issue, considering that the agents are able to solve the problem of a fire emergency. In a real-world situation, this type of behavior could be implemented using agents with smoke sensors and using communication mechanisms to enforce the established social conventions.