Robot Waste Mission

Multi-Agent Coordination for Hazardous Waste Cleanup

Project carried out by Group 3:
Oumaima CHATER, Laure-Emilie MARTIN, Agathe PLU, Agathe POULAIN

Report written by: Oumaima CHATER

CentraleSupélec

April 2024

For full project code and data:
`github.com/Ouma487/Robot_Mission`

Contents

1 Introduction

This project explores a multi-agent system designed to clean up radioactive waste in a simulated environment. The environment is split into three zones with increasing radioactivity, each subject to specific constraints on robot operations. Robots have to coordinate to pick up, transform, and finally dispose of the waste, passing it along a pipeline that demands careful sequencing.

Initially, we wanted to study how such systems behave without any form of coordination. Later, we investigated how introducing communication, task assignment, and hierarchy could improve overall efficiency and help avoid deadlocks.

2 Project Overview

2.1 Environment and Objectives

The environment is implemented as a 2D grid divided into three zones. The leftmost area (Zone 1) contains green wastes and has low radioactivity. The middle area (Zone 2) has medium radioactivity and yellow wastes, while the rightmost part (Zone 3) is highly radioactive and serves as the disposal zone for the final red wastes.

The workflow of cleaning is inherently staged: green agents work in Zone 1 to pick up two green wastes and transform them into a yellow waste, which they then carry to the border with Zone 2. Yellow agents do something similar in Zones 1 and 2, transforming yellow wastes into red ones. Finally, red agents traverse all zones to collect red wastes and bring them to the disposal zone.

This pipeline structure forces the robots to rely on each other indirectly, even before adding explicit communication. The primary goal is to minimize redundant trips and avoid situations where agents get stuck holding partial loads they cannot complete.

3 System Architecture

3.1 Initial Architecture Without Communication

Our first implementation gave each agent only local rules. Robots moved randomly within their allowed zones, searching for waste to pick up. When they had enough items, they transformed them and moved to drop them off at the next stage. This approach did manage to eventually clean the grid in some cases, but frequently resulted in deadlocks. For instance, two agents might each pick up one of the last pair needed for a transformation, preventing either from finishing their task.

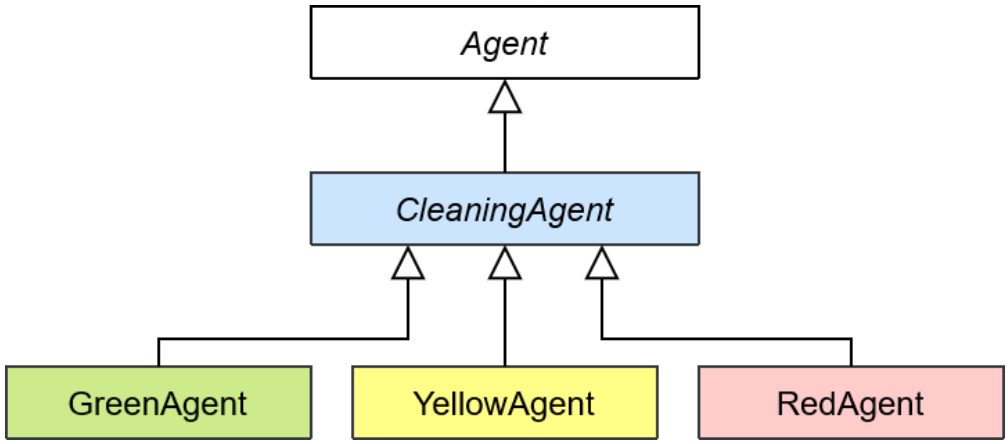

Figure 1: Initial UML diagram without communication: agents operated independently.

3.2 Extended Architecture With Chiefs

To address these inefficiencies, we added a simple hierarchical communication system. Each group of agents (green, yellow, or red) is assigned a chief. These chiefs maintain a global view of their respective zones by aggregating information sent by the agents. They also coordinate with each other to hand off transformed waste, ensuring a smoother workflow.

This change meant agents were no longer operating purely on local observations but followed instructions from their chiefs, who could plan globally.

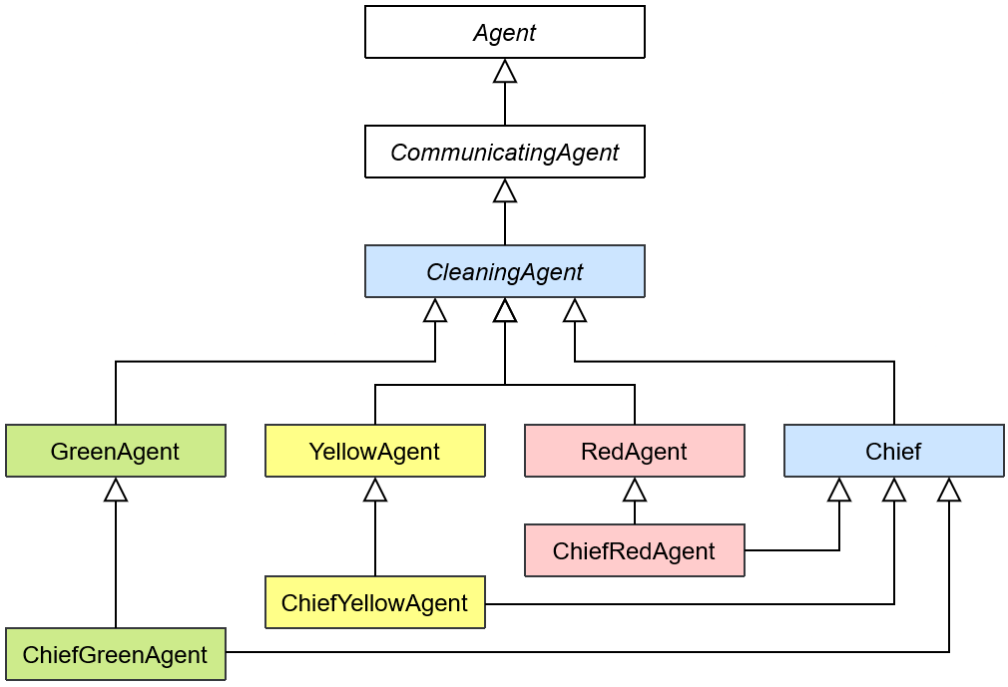

Figure 2: Extended UML with chiefs coordinating agents and exchanging information.

4 Communication and Coverage Strategy

4.1 Initial Coverage Phase

One of the improvements introduced with chiefs was a systematic coverage phase. Rather than moving randomly, agents initially sweep through rows of their zones under the chief's direction. This ensures that all wastes are detected early on, allowing the chiefs to build a complete map of where items are located.

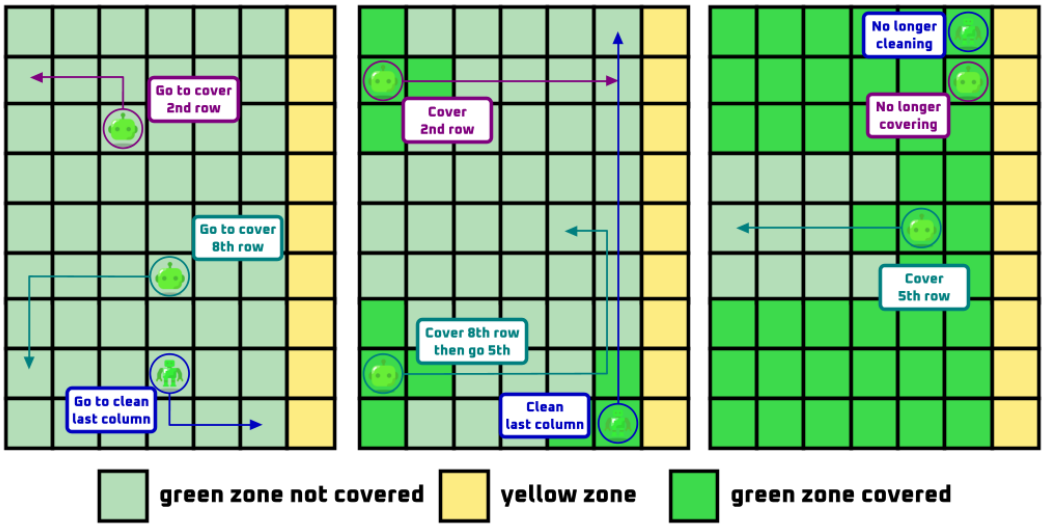

Figure 3: Early coverage phase where agents explore rows systematically.

4.2 Dynamic Task Allocation

Once the chiefs have this information, they start assigning agents to specific targets. If an agent happens to pick up waste along the way, the chief updates plans accordingly. This flexibility avoids duplicated work and also prevents classic problems like two agents trying to reach the same piece of waste.

By managing both the initial exploration and subsequent assignments, the chiefs help maintain a balanced workload and reduce unnecessary movements.

5 Results and Interpretation

We ran multiple experiments comparing the system with and without communication. The figure below summarizes how many steps were needed on average to clean the entire map, as well as the volume of messages exchanged in the coordinated approach.

	Without communication	With communication	
	Steps	Steps	Messages
1 agent	1346	367	204
2 agents	713	212	700
3 agents	404	162	857
4 agents	NA	131	1015

Figure 4: Average steps and number of messages for different numbers of agents.

Without any coordination, adding more agents generally reduced the number of steps — up to a point. However, it also significantly increased the risk of deadlocks, especially when three or four agents were deployed per zone. Many runs simply failed to complete because agents ended up blocking each other with partial loads.

With the introduction of chiefs and a lightweight communication protocol, we observed a dramatic improvement. The cleaning process completed roughly three to four times faster on average. More importantly, this approach completely eliminated deadlocks. The tradeoff was clear: while the system moved more efficiently, it required sending hundreds or even thousands of coordination messages in the largest setups.

6 Conclusion and Future Directions

This project illustrates how even relatively simple local agents can collectively achieve complex tasks when guided by a minimal layer of coordination. By introducing chiefs that handle both spatial exploration and task sequencing, we effectively overcame the bottlenecks and deadlocks inherent to purely decentralized approaches.

Looking ahead, it would be interesting to simulate scenarios where communication is unreliable, such as dropping some messages at random, to see how robust our approach remains. Another natural extension would be introducing agent fatigue or failures due to prolonged exposure to radioactivity, requiring chiefs to adapt by reassigning tasks or electing new leaders.

These ideas could easily be applied beyond this academic exercise to many multi-stage pipeline problems in logistics or swarm robotics.