Project Proposal of Multi-Agent Cleaning Robots

Duarte Silva 93234 Gonçalo Mesquita 94196 Laura Baeta 92507

ABSTRACT

This project presents a multi-agent system for coordinating the actions of multiple Roomba-like robots to efficiently clean a house or office. The primary objective is to design a decentralized system where the robots cooperate, negotiate, and adapt their strategies to achieve optimal cleaning performance in complex environments. The project involves defining goals, environment parameters, communication and negotiation protocols, decision-making processes, and evaluation metrics. By integrating peer-to-peer communication, path planning algorithms, conflict resolution techniques, and reinforcement learning, the agents are expected to learn and adapt their coordination strategies. The main contributions include the development of an efficient, adaptable multi-agent cleaning system and identifying optimal strategies for cleaning performance optimization through empirical evaluation. These results can enhance the performance of cleaning robots in real-world applications, making them more suitable for various settings.

1 INTRODUCTION

1.1 Motivation

Cleaning is an essential yet time-consuming task in both homes and offices. The advent of robotic vacuum cleaners, such as Roombas, has offered a more convenient and efficient way to maintain cleanliness. However, current Roomba implementations usually involve a single unit that operates independently, which may not be optimal for larger or more complex environments. A multiagent system of Roombas can potentially enhance the cleaning process by coordinating their efforts and addressing different cleaning tasks concurrently.

1.2 Related work

The use of a multi-robotic system offers several advantages, primarily due to its ability to perform tasks that a single robot may not be able to complete or would take much longer to accomplish. Several examples of these benefits are highlighted in sources[1]-[4]. For instance, in source [3], a vacuum robot uses the rapidly exploring random tree (RRT) technique for navigation and path planning, allowing for efficient large-scale indoor vacuum cleaning. Additionally, in source [1], Connell and La utilize the RRTx algorithm for path replanning in a non-static environment, demonstrating that the multi-robot approach generates equivalent or better paths while also being time-efficient.

1.3 Problem definition and Relevance

The project aims to develop a multiagent system of Roombas that can efficiently clean a house or an office environment. The system should be capable of autonomously coordinating Roombas to complete cleaning tasks while avoiding conflicts and ensuring optimal coverage. This project is relevant as it addresses a real-world problem, where efficient cleaning is crucial for maintaining a healthy and comfortable living or working space.

AASMA Project, IST, 2023, Lisbon, Portugal

https://doi.org/

1.4 Objectives

- Develop a multiagent system of Roombas that can adapt to various environments, such as homes and offices, with different layouts and cleaning requirements.
- Investigate different decision-making strategies, excluding centralized control, for efficient coordination among Roombas, considering factors such as task allocation, navigation, and resource management.
- Evaluate the performance of the multiagent Roomba system in terms of cleaning efficiency, time, and coverage, as well as the ability to handle conflicts and complex coordination problems.
- Compare the effectiveness of the proposed multiagent Roomba system against traditional single-agent Roomba implementations.

Cleaning a house is a time-consuming and tedious task, but it is essential to maintain a healthy living environment. The development of Roombas has made it easier to clean a house without much human intervention, but it is still not efficient enough. In this project, we aim to implement a multi agent system that coordinates the actions of multi agent Roombas to clean a house as quickly and efficiently as possible while cooperating and negotiating with each other. The motivation behind this project is to demonstrate the potential of multi agent systems in solving real-world problems, such as cleaning a house.

2 APPROACH

To address the problem we have to define our multi agent system, our environment and system architecture. Inside the system architecture we will specify various techniques, including perception, decision making (planning and learning), negotiation, and communication.

The environment for the multi agent Roomba system will consist of simulated house or office spaces with varying layouts, obstacles, and cleaning requirements. These environments will include rooms of different sizes and shapes and objects that may complicate the Roombas' task. The environments will also be characterized by different levels of dirt distribution and types, which the Roombas will need to clean effectively. There will be two types of dirt necessitates, one Roomba can effectively clean one type of dirt while the other type requires the coordinated effort of two Roombas to achieve complete cleanliness. The objective is to try several maps in order to increase complexity of the problem.

In our project, we will test various types of agents to increase the complexity, such as the reactive agent, the hybrid agent, and if we have the time, a reinforcement learning agent. The multi-agent system will comprise multiple Roomba agents, each equipped with sensors and actuators to perceive and interact with the environment. The Roombas will have the ability to detect obstacles, dirt, and other agents while navigating and performing cleaning tasks. Each Roomba agent will be capable of making autonomous decisions based on its perception of the environment and the state of other agents.

The proposed multi agent Roomba system will follow a decentralized architecture, where each Roomba agent operates independently, without a central controller. The Roombas will communicate and coordinate with each other through a peer-to-peer communication mechanism, exchanging information about their current state and locations. This architecture allows for greater flexibility and adaptability to different environments and situations.

The approach will involve the investigation and implementation of different decision-making strategies, such as:

1.Navigation and negotiation: Roombas might use reinforcement learning methods such as Q-Learning or policy based methods to decide their actions and coordination between the multi agents.

2. Navigation: Roombas will employ path-planning algorithms, such as A*, to find optimal paths for cleaning tasks while avoiding obstacles and collisions with other agents.

3. Conflict Resolution: In situations where Roombas have conflicting goals or face complex coordination problems, they will use negotiation, consensus, or other conflict resolution techniques to determine the best course of action and ensure efficient cleaning.

To explain why the design choices are adequate to address the problem of efficient cleaning in complex environments using a multi agent Roomba system, we can go through each choice step by step.

A decentralized architecture allows each Roomba agent to operate independently without a central controller. Since we are going to change the environment complexity, the decentralized architecture promotes flexibility and adaptability.

Implementing a peer-to-peer communication mechanism allows Roomba agents to exchange information about their current state and location. This design choice helps the agents coordinate their actions more efficiently and avoid conflicts, leading to improved cleaning performance.

Implementing path-planning algorithms like A* helps Roomba agents find optimal paths for cleaning tasks while avoiding obstacles and collisions with other agents. This design choice ensures that agents navigate the environment effectively and safely, contributing to the system's overall cleaning performance. As well as it might be a good e empirical evaluation comparing to our reinforcement learning results.

Incorporating reinforcement learning enables Roomba agents to plan, learn and adapt their coordination strategies based on their experiences in the environment. This design choice allows the agents to continuously improve their performance and adapt to our different environments or cleaning distributions. Making the overall system more robust.

By incorporating these design choices into the multi agent Roomba system, the agents can effectively coordinate their actions, adapt to different environments and situations, and optimize their cleaning performance, addressing the problem of efficient cleaning in complex environments.

3 EMPIRICAL EVALUATION

In order to empirically evaluate the multi-agent Roomba system, it is essential to establish a comprehensive set of metrics that will allow us to verify the accomplishment of the project's objectives. For this we defined the following metrics.

The first metric is **cleaning efficiency**, which quantifies the proportion of dirt removed by the Roomba agents in relation to the total amount of dirt present in the environment. This metric serves to validate the system's primary objective of effectual environmental cleaning.

The second metric is **cleaning time**, which records the time required by the Roomba agents to accomplish the cleaning tasks across various environments. This metric facilitates the examination of the efficiency of coordination strategies and the influence of diverse approaches on overall cleaning performance.

The third metric is **coverage**, which computes the percentage of the environment's floor space traveled by the Roomba agents throughout the cleaning process. This metric makes sure that the agents navigate and explore the environment efficiently to locate and clean the dirt.

The fourth metric is **collisions**, which enumerates the occurrences of collisions between Roomba agents and between Roomba agents and environmental obstacles. This metric assesses the efficacy of navigation and coordination strategies in preventing conflicts.

Finally, the fifth metric is **adaptability**, which investigates the Roomba agents' capacity to adapt to alterations in the environment or novel cleaning requirements by quantifying the time needed to modify their strategies and the effects of these modifications on cleaning performance.

By using these metrics during the empirical evaluation of the multiagent Roomba system, we can evaluate the effectiveness of design choices, coordination strategies, and the reinforcement learning approach in addressing the challenge of efficient cleaning in complex environments. The comparison of different approach outcomes will help identifying optimal strategies for enhancing cleaning performance and adaptability.

REFERENCES

[1] Devin Connell and Hung Manh La. "Extended rapidly exploring random tree–based dynamic path planning and replanning for mobile robots". en. In: *International Journal of Advanced Robotic Systems* 15.3 (May 2018), p. 172988141877387. ISSN: 1729-8814, 1729-8814. DOI: 10.1177/1729881418773874. URL: http://journals.sagepub.com/doi/10.1177/1729881418773874 (visited on 05/09/2023).

[2] Long Jin et al. "Dynamic task allocation in multi-robot coordination for moving target tracking: A distributed approach". en. In: *Automatica* 100 (Feb. 2019), pp. 75–81. ISSN: 0005-1098. DOI: 10.1016/j.automatica.2018.11.001. URL: https://www.sciencedirect.com/science/article/pii/S0005109818305338 (visited on 05/09/2023).

[3] Agris Nikitenko et al. "Multi-robot System for Vacuum Cleaning Domain". In: *Advances in Practical Applications of Heterogeneous Multi-Agent Systems. The PAAMS Collection*. Ed. by David Hutchison et al. Vol. 8473. Series Title: Lecture Notes in Computer Science. Cham: Springer International Publishing, 2014, pp. 363–366. ISBN: 978-3-319-07550-1 978-3-319-07551-8. DOI: 10.1007/978-3-319-07551-8_39. URL: http://link.springer.com/10.1007/978-3-319-07551-8_39 (visited on 05/09/2023).

[4] Zachary Young and Hung Manh La. "Consensus, cooperative learning, and flocking for multiagent predator avoidance". en. In: *International Journal of Advanced Robotic Systems* 17.5 (Sept. 2020), p. 172988142096034. ISSN: 1729-8814, 1729-8814. DOI: 10.1177/1729881420960342. URL: http://journals.sagepub.com/doi/10.1177/1729881420960342 (visited on 05/09/2023).