MIND THE GAP

# Autonomous Collaborative Ground Traversal of Discontinuous Terrain

## Conceptual Design Report

**TEAM A**

Dhanvi Sreenivasan | Hari Shankar Ravisankaran | Sandhya Vidyashankar | Sankalp Chopkar | Sudhansh Yelishetty

**Sponsors:** Prof. Zeynep Temel | Prof. Katia Sycara | Sha Yi

**Date:** December 13, 2022

Master of Science Robotic Systems Development | Carnegie Mellon University The Robotics Institute

## Table of Contents:

## 1. Project Description

Robot swarms have been shown to improve the ability of individual robots by inter-robot collaboration. Introducing physical coupling between robots can further extend their ability in an unstructured environment. This is particularly beneficial in inspection scenarios, especially in mines, where the environment is complicated and thus challenging. Instead of having the crew spend precious day hours inspecting several sites in a mine in person, they can deploy a swarm of robots to do the same in a more efficient and collaborative manner.

Rather than sending in a team of people for inspection at each site, it is more efficient to send a swarm of robots to collect more information at each point of interest to the crew. Each agent in the system will inspect a set of allocated points of interest, and once every point is done, they will return all the information to the end users, using which further plans can be made.

Previously, PuzzleBots [1][2] was proposed for the same. It is a robotics swarm system where the robots can dynamically couple with each other to form bridges and decouple to perform individual tasks but on a small scale. This system can either be used to build bridges for the transportation of other materials, or to get the robots from one side of a gap to the other. The aim of this project is to investigate the physical coupling of robot swarms on a larger scale, as well as implement and test existing control and planning algorithms on a larger system.

## 2. Use Case

Consider a mine with several hotspots, where a hotspot is a location with high concentrations of mineral deposits. These minerals are of critical importance and the mining crew is on a time crunch. That being said, each hotspot in the mine must be inspected in detail before approaching the next steps.

Mines, in general, can be narrow, in caves, and can have cracks or fissures on the ground or other surfaces. Due to rugged terrains in such unstructured environments, traversing these conditions can be potentially risky for humans, e.g., collapse of overhead rock structures. The mining crew needs to inspect each hotspot but will lose out on time and need to find a better solution. An autonomous distributed solution.

Bring in PuzzleBots. The crew inputs a blueprint of the map of the mine. This would typically be an occupancy grid of the mine with hotspots marked in them. All agents in the system start from the base station at the entrance of the mine. Each agent gets allocated hotspots optimally, and collision-free optimal trajectories are planned. The agents branch off from the swarm to inspect individual hotspots. In case of emergencies involving imminent collisions, each agent is equipped with a collision avoidance system. They spend a predetermined fixed amount of time at the hotspots before being allocated new ones.

It is very likely that hotspots are located across fissures in the mine, which will be referred to as gaps. Should there be any hotspots on either side of any gaps, the system determines the feasibility of getting there. If feasible, the system plans for a group of agents to physically couple with each other and cross the gap. After that, they split up, and follow their respective planned trajectories to inspect individual hotspots. After inspecting the entire map, the agents return to the base station.

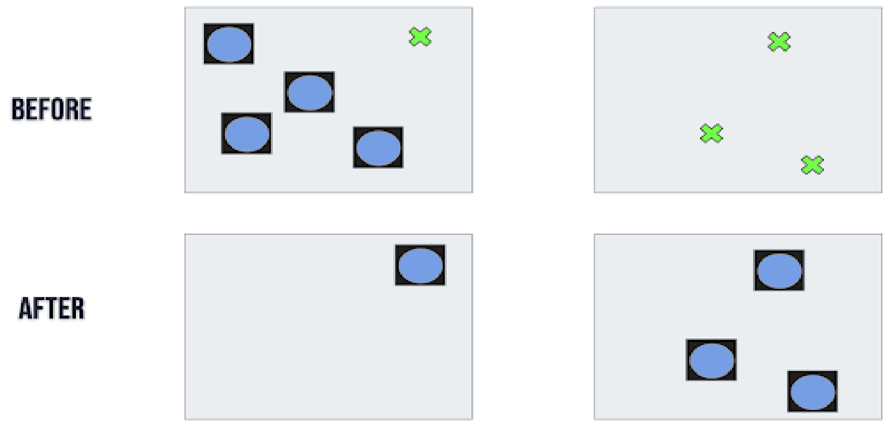


**Figure 1. Use Case Illustration**

## 3. System-Level Requirements

Our system-level requirements were elicited from the use case, discussions with our sponsors, and internal team meetings based on the time and budget constraints to the project timeline. All the requirements are shown in Table 1 to Table 3.

### 3.1 Mandatory Requirements

#### 3.1.1 Functional Requirements

**Table 1. Functional Requirements**

| | Requirement |
|---|---|
| MF1 | The system shall localize agents in a given map. |
| MF2 | The system shall route agents and avoid collisions. |
| MF3 | The system shall  determine feasible gaps. |
| MF4 | The system shall determine and achieve coupled configurations. |
| MF5 | The system shall cross gaps. |
| MF6 | The system shall reach given regions of interest. |

### 3.1.2 Performance Requirements

**Table 2. Performance Requirements**

| | Requirement |
|---|---|
| MP1 | The system will cross gaps up to 1.5 agent lengths. |
| MP2 | The system will have 0 unplanned collisions between agents. |
| MP3 | The system will achieve formations with at least 3 robots. |
| MP4 | The system will cross feasible gaps 75% of the time. |
| MP5 | The system will reach all POIs 75% of the time. |
| MP6 | The coupling mechanism will bear the weight of one agent. |
| MP7 | The coupling mechanism will self-align against heading errors of 5 degrees and position error of 10 mm. |

### 3.1.3 Non-Functional Requirements

**Table 3. Non-Functional Requirements**

| | Requirement |
|---|---|
| MN1 | The weight of an agent shall be minimal. |
| MN2 | The coupling mechanism shall consume a low amount of energy. |
| MN3 | The system shall be scalable. |
| MN4 | The system shall be easily maintainable. |
| MN5 | The team shall maximize learning and fun throughout the project through flexible work plans across subsystems. |

### 3.2 Desirable Requirements

We have chosen not to go with any desirable requirements because the project is highly complex.

## 4. Functional Architecture

### 4.1 System Architecture

The system is divided into two blocks: the server and the agents. A brief overview of the functions of agents and the server is shown below:

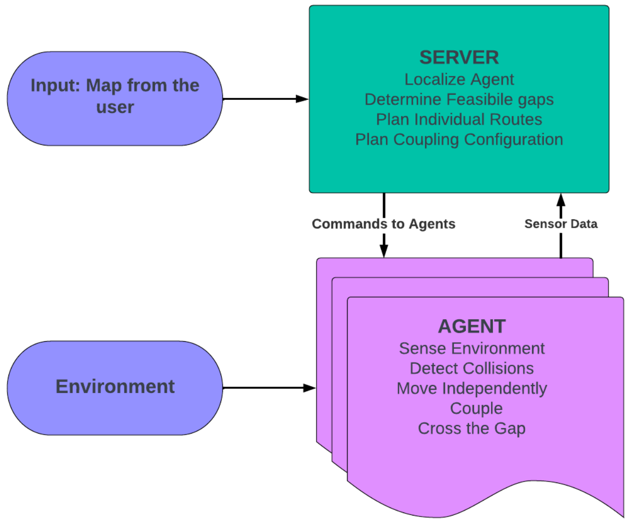


**Figure 2. Functional Architecture of the System**

## 4.2 Server functional architecture

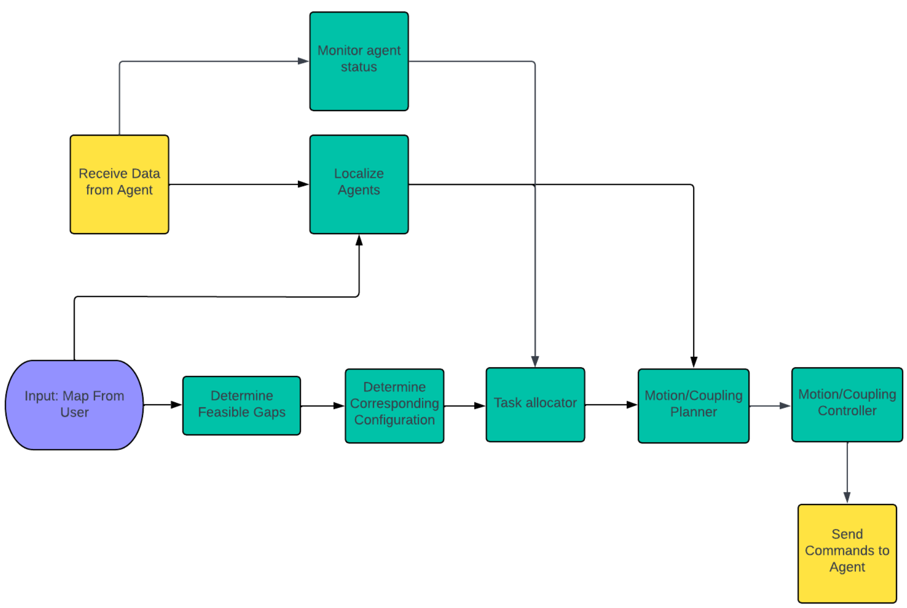


**Figure 3. Functional Architecture of the Server**

The user provides an initial map of the environment to the server which includes information about the gaps, the points of interest, and the initial agent poses. The server receives sensor data from the agents. The server monitors the agent's status and localizes each agent within the environment. Based on the input map, the server determines the feasible gaps and also the configuration required to cross the gaps. The task allocator takes in this information and allocates specific tasks to each individual agent.

The tasks assigned are then sent to the planner block which generates either collision-free paths for agents to cover the points of interest or coupling and gap-crossing plans if the task is to cross the gaps. To execute the planned paths, the controller block computes the error between the desired and current agent pose to generate linear and angular velocity commands for movement and coupling commands to enable coupling.

## 4.3 Agent functional architecture

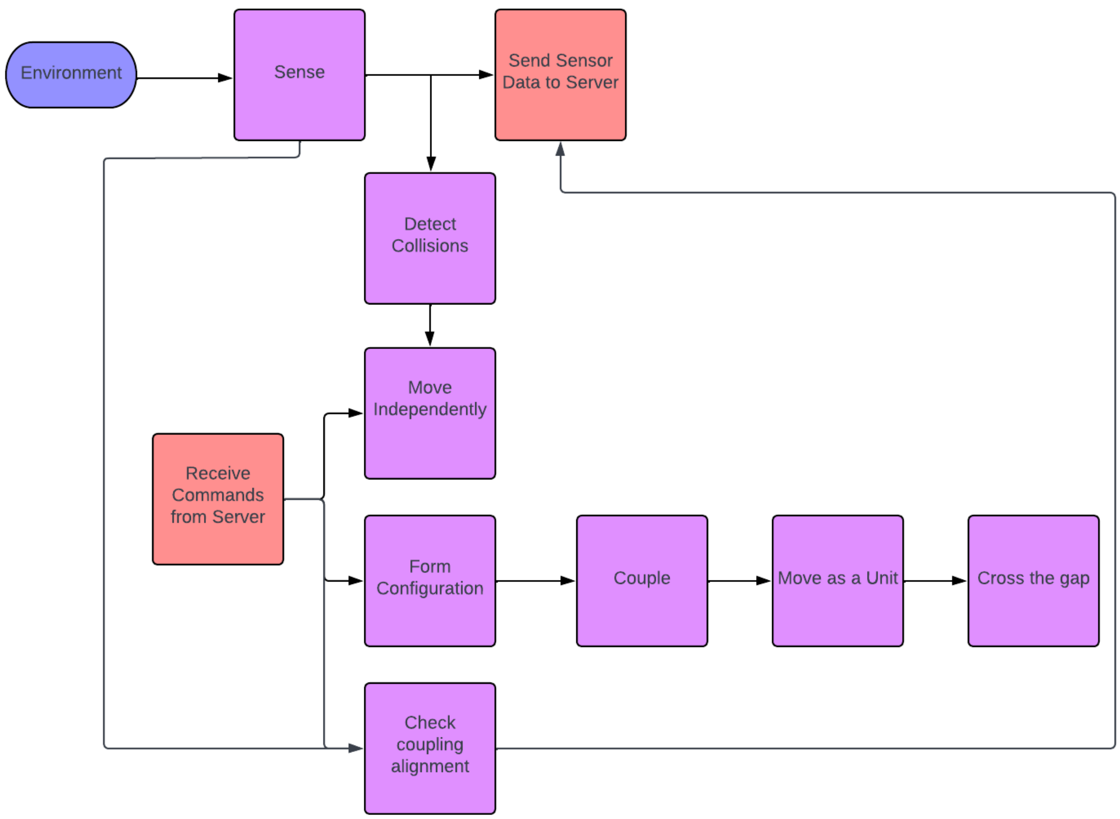


**Figure 4. Functional Architecture of the System**

The agent receives linear, angular velocities, and coupling commands from the server. The linear and angular velocities are used to generate the motor commands that enable the agent to move independently to cover points of interest, form configuration, and move as a unit depending upon the state of the agent. When the agent moves, the sensor readings are continuously transmitted to the server. The sensor data is also used to detect imminent collisions and avoid them. The coupling commands are used to actuate the coupling motors to enable successful coupling between the agents. To refine localization estimates for coupling, the agent also performs coupling alignment checks using the sensor readings which are then sent to the server.

# 5. System and Subsystem Level Trade Studies

## 5.1. System Level Trade Study

This high-level trade study contrasts several possibilities for gap crossing robots. Each system (legged robot, drone etc) has its own advantages and disadvantages. Keeping in mind the mine exploration use case, we have weighed each solution to select the coupled multi robot swarm.

**Table 4: System-level trade study**

| Solution | Variable Length | Jumping | Drones | Coupled Swarm |
|---|---|---|---|---|
| Favorable | Simple Controls | Fast<br><br>No constraints on the gap | Fastest<br><br>No constraint on gap<br>Self sufficient | No constraints on gap<br><br>Low work volume<br><br>Variable configurations |
| Unfavorable | Weight Distribution<br><br>Maintaining Balance<br><br>Life Cycle<br><br>Constraints on gap | Vertical Clearance<br><br>Difficult to control<br><br>Mechanical wear and tear<br><br>Payload constraints | Vertical clearance<br><br>Susceptible to weather<br><br>Inefficient for ground tasks | Slow<br><br>Difficult to coordinate |

## 5.2 Mobility Platform Trade Study

The choice of mobile robot platform is essential to the overall success of the project. We have chosen to purchase an existing platform rather than make one from scratch, keeping in mind factors including cost, time constraints and repeatability. Instead, we will be building an enclosure for an existing platform that will house the coupling mechanism. Keeping this in mind, we collected data on 4 existing platforms, summarized in the end of the document**.** The Khepera and the Roomba are two robot platforms that are already available to us. The E-Puck and the Hero-Bot are robots developed specifically for swarm robotics research with sufficient compute and sensing for our project

The original Puzzlebot paper proposed a custom platform of dimensions of approximately 50x50x50mm. It is therefore necessary to choose a mobile base that is sufficiently larger to satisfy the requirement of scaling up the original robots. It is also highly preferable that the system has a sufficiently low weight and high payload to allow for more freedom in the coupling mechanism design.

Other factors considered in this trade study include cost, ease of use and user customizability.

We used this information in the trade study, presented in Table 5**.** The highest weight was assigned to payload bearing capabilities, weight and size.

**Table 5: Mobility Platform Trade Study**

| **Options** | | | **Khepera** | **E-Puck** | **Hero 2.0** | **Roomba** |
|---|---|---|---|---|---|---|
| **Criteria** | | **Weights** | Scores (scale from 0-5) | | | |
| Payload | | 20 | 3 | 4 | 2 | 5 |
| Weight | | 15 | 4 | 5 | 5 | 1 |
| Size | | 15 | 5 | 0 | 0 | 2 |
| Motor torque | | 10 | 2 | 1 | 1 | 5 |
| Ease of Use | | 10 | 4 | 3 | 5 | 3 |
| Sensing | | 10 | 5 | 4 | 4 | 3 |
| Manoeuvrability | | 10 | 1 | 3 | 3 | 3 |
| Cost | | 5 | 5 | 0 | 2 | 5 |
| Customization | | 5 | 3 | 4 | 4 | 2 |
| Cumulative Scores | | 100 | 3.55 | 2.85 | 2.75 | 3.2 |

## 5.3. Coupling mechanism trade study

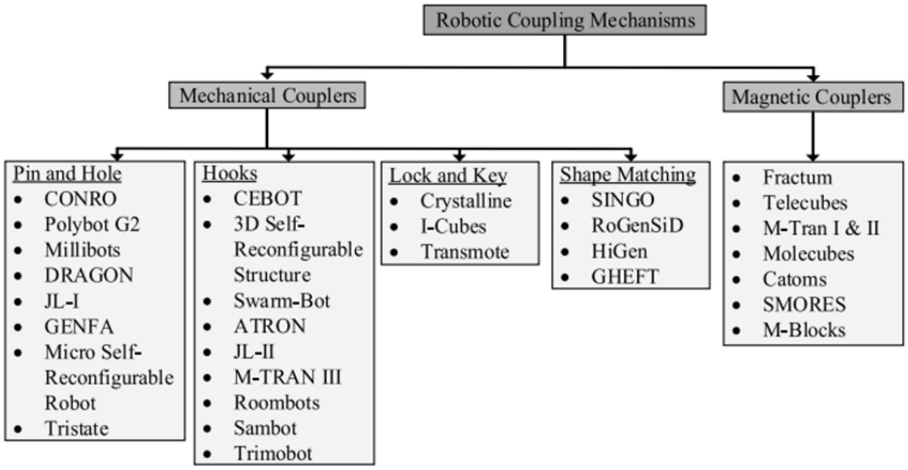


**Figure 5. Summary of Modular Coupling Mechanisms**

In the trade study, we have initially compared the two main coupling categories (mechanical and magnetic coupling) and then proceeded to contrast the various kinds of mechanical coupling, listed above, on the same factors.

**Table 6: Overall Coupling Mechanism Type Trade Study**

| | | **Mechanical** | **Magnetic** |
|---|---|---|---|
| Parameters | Weights | Scores (scale of 1 - 5) | |
| Vertical Load Bearing | 30 | 4 | 3 |
| Lateral Force transmission | 15 | 3 | 2 |
| Misalignment Tolerance | 15 | 3 | 5 |
| Weight | 15 | 4 | 1 |
| Power Consumption | 10 | 3 | 5 |
| Cost | 5 | 4 | 2 |
| Ease of Manufacturing | 5 | 3 | 5 |
| Replaceability | 5 | 4 | 2 |
| Cumulative Scores | 100 | 3.55 | 3.05 |

It is important that the coupling mechanism is able to bear sufficient load as well as has some tolerance for misalignment to make up for inaccuracies in localization. As we intend for the robots to locomote as a unit after coupling, sufficient transmission of lateral forces also becomes necessary to maintain the configuration.

Other considerations include cost, repeatability of the design, manufacturing complexity and power consumption. It will be important to make the mechanism as light as possible and simple enough that parts can be manufactured and replaced quickly as the need arises.

**Table 7: Mechanical Coupling Mechanisms Trade Study**

| **Options** | | **Pin and Hole** | **Hooks** | **Lock and Key** | **Shape Matching** |
|---|---|---|---|---|---|
| **Parameters** | **Weights** | Scores (scale from 1-5) | | | |
| Vertical Load Bearing | 30 | 4 | 2 | 5 | 2 |
| Lateral Force transmission | 15 | 3 | 4 | 3 | 1 |
| Misalignment Tolerance | 15 | 3 | 4 | 1 | 2 |
| Weight | 15 | 3 | 2 | 3 | 5 |
| Power Consumption | 10 | 4 | 2 | 4 | 5 |
| Cost | 5 | 5 | 2 | 2 | 3 |
| Ease of Manufacturing | 5 | 3 | 4 | 2 | 1 |
| Replaceability | 5 | 4 | 5 | 3 | 1 |
| Cumulative Scores | 100 | 3.55 | 2.85 | 3.3 | 2.55 |

# 6. Cyber-physical architecture

## 6.1 Server cyber-physical architecture

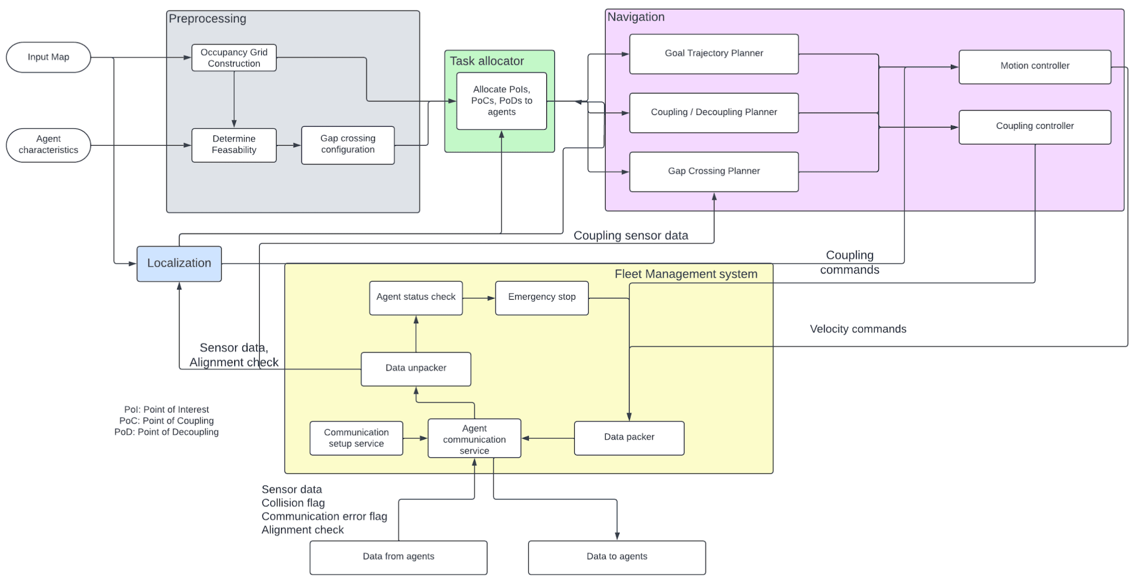


**Figure 10. Server Cyber-Physical Architecture**

The server acts as the brain of the system and controls the behavior of the individual agents. It takes an initial map of the environment and a config file from the user. The map contains information about the gaps and points of interest. The config file contains information about all the available agents. The FMS block uses the config file to set up communication service between the agents and the server. FMS handles the packing of data that needs to be sent to the agents and the unpacking of data from the agents to feed them into the corresponding subsystems. The map and the sensor information are used by the localization subsystem to determine the poses of each agent. The FMS also monitors the individual agents to check for communication failures.

The map provided by the user is preprocessed into Voronoi diagrams/connected grids to feed into the task allocation block. The preprocessing block also determines the feasible gaps in the map and the corresponding configuration required to cross those gaps. Based on these, the task allocator assigns tasks to each individual agent. The tasks can be covering points of interest or reach points of coupling and decoupling. Allocation is formulated as a modified Multiple Traveling Salesman problem to account for the gaps and coupling behavior.

The assigned tasks are then passed to the navigation subsystem. The navigation subsystem includes two major blocks: the trajectory planner and the controller. Based on the assigned tasks, for each agent, the goal trajectory planner will be used to generate collision-free paths to cover points of interest, the coupling planner will be used to generate coupling paths and commands, the gap crossing planner will be used to generate paths for the coupled robots to cross the gaps. The controller takes in the generated paths and the actual agent pose to compute the desired linear, angular velocities for agent movement and coupling commands for enabling coupling. These commands are then passed to the FMS to send to the agents.

## 6.2 Agent cyber physical architecture

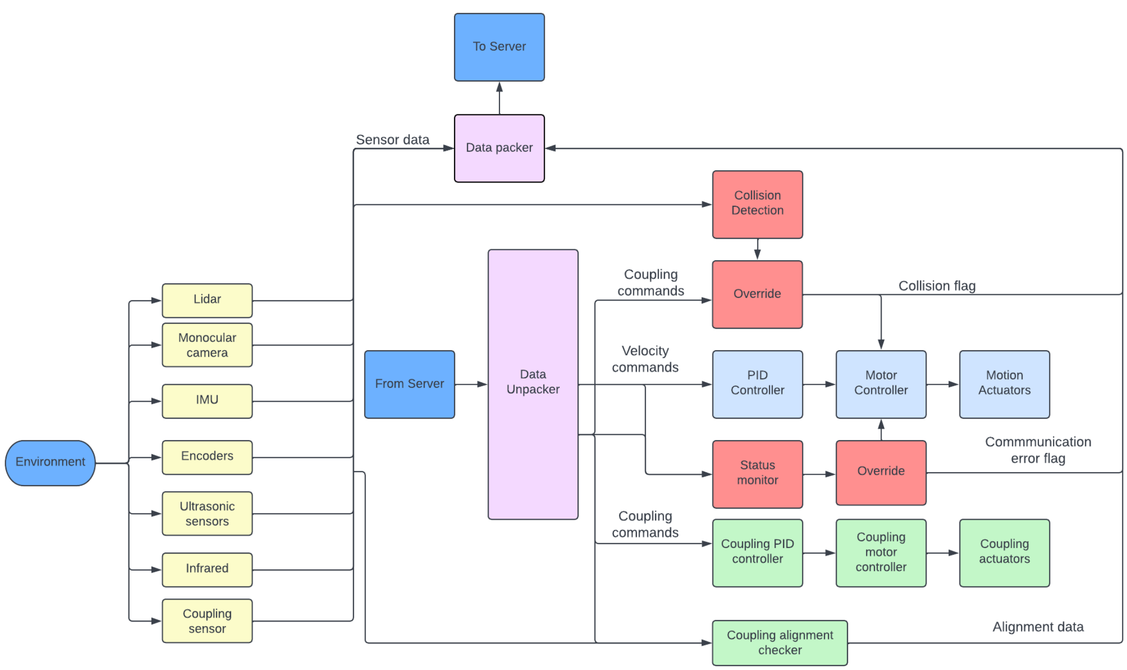


**Figure 11. Agent Cyber-Physical Architecture**

Based on the trade studies, we are using Khepera 4 robots for the agents. The platform was primarily designed for testing multi-agent systems, so they are suitable for our purpose. The sensors include wheel encoders, a 3-axis gyroscope, a 3-axis accelerometer, infrared sensors, ultrasonic sensors, laser range finders, RGB camera. All of these sensor data are packaged and transmitted to the server. The agent runs a collision detection service on its own to detect imminent collisions and avoid them. It also monitors its own status based on the server commands and sends messages to the server in case of any failure. The velocity and coupling commands from the server are passed to the respective controllers which convert them to actuator velocities. For refinement of localization estimates during coupling, the agent also runs an alignment-checking algorithm based on Aruco markers and sends the processed data to the server.

# 7. Subsystem Description

## 7.1 Coupling Mechanism

This subsystem is a hardware addition to the robot mobility platform Khepera IV. It is responsible for facilitating the physical coupling between robot agents, allowing them to collaborate and traverse discontinuous terrain. The coupling mechanism uses a pin-and-hole method of mechanical coupling and is developed from electromechanical components like a Linear actuator to actuate the pin, a Solenoid Lock to latch on to the pin after complete actuation, Roller support to facilitate the sliding of the pin, Microcontroller in addition to Khepera Compute system and Battery as per power requirements. The coupling mechanism is strong enough to support the weight of another robot and is designed to compensate for minor misalignments.

## 7.2 Localization

The pose of every agent on the map needs to be determined precisely to enable coupling between the robots and cover the points of interest efficiently. The user provides a map and the initial positions of the agents. The sensor readings from the agent will be used for getting the pose estimate. The wheel encoders provide an initial first estimate of the robot's pose. Then the data from the laser range finder is used to refine the estimate of the change in the pose by matching scans of consecutive frames. The laser scans are also matched with the provided map. Fusion of these data will be done using Extended Kalman or particle filters. This process is done separately for each agent using the map provided by the user.

During the coupling process, the localization precision should be high. Therefore, when the robots are sufficiently close to each other during coupling, additional redundant methods like Aruco marker-based pose estimation will also be used along with the previously mentioned method.

## 7.3 Preprocessing

The preprocessing block involves the processing of the map into Voronoi diagrams/connected grids or any other format required by the planning block. Additionally, the map also contains information about the gaps in the system. The preprocessing block determines all the feasible gaps in the map based on the number of agents available and also determines the configuration required by the agents to cross those feasible gaps. The infeasible gaps will be considered obstacles in the map. The outputs from the system include the processed map, feasible gaps, and their corresponding configuration.

## 7.4 Task allocation

The task allocator assigns uncompleted tasks to each agent based on the inputs from the preprocessing block while minimizing the overall time taken. Specifically, it will allocate points of interest to cover each agent and also allocate the points of coupling and decoupling in case the agent has to cross a gap. Greedy heuristics or some sort of optimization technique will be used for this purpose. The problem will be formulated as a modification of Multiple Traveling Salesman to include the gaps and coupling behaviors. It outputs the next task for each agent and sends it to the navigation subsystem.

## 7.5 Navigation

The navigation subsystem includes a trajectory planner and controller. The trajectory planner is further categorized as a goal trajectory planner, coupling planner, and gap-crossing planner. Based on the inputs from the task allocator and the localization blocks, one of the three planners will be used to generate collision-free paths to cover points of interest, coupling strategies, or gap-crossing paths. The high-level controller in the server then computes the error in the actual and desired pose for each agent and generates desired linear, angular velocities or coupling commands depending upon the assigned tasks. Based on these commands, the low-level controller of the Khepera platform will convert these into desired actuator velocities.

## 7.6 Fleet management system

The fleet management system serves as the communication bridge between all the agents and the server. The user provides a config file with information about all the agents in the fleet to set up communication channels for transmission and reception for all the agents. It also handles packing, unpacking, and logging of data and monitors the agents during operation. It relays information back to all the other subsystems in case of any changes in the fleet.

# 8. Project Management

## 8.1 Work Breakdown Structure

We choose to split the project into 5 key verticals - Hardware, Fleet Management System, Planning, Localization and Management. This is derived from the major blocks in our cyber physical architectures explained above (section ref.).

Designing, testing and integrating units from the WBS form our key milestones for the upcoming semester, according to the plan detailed below

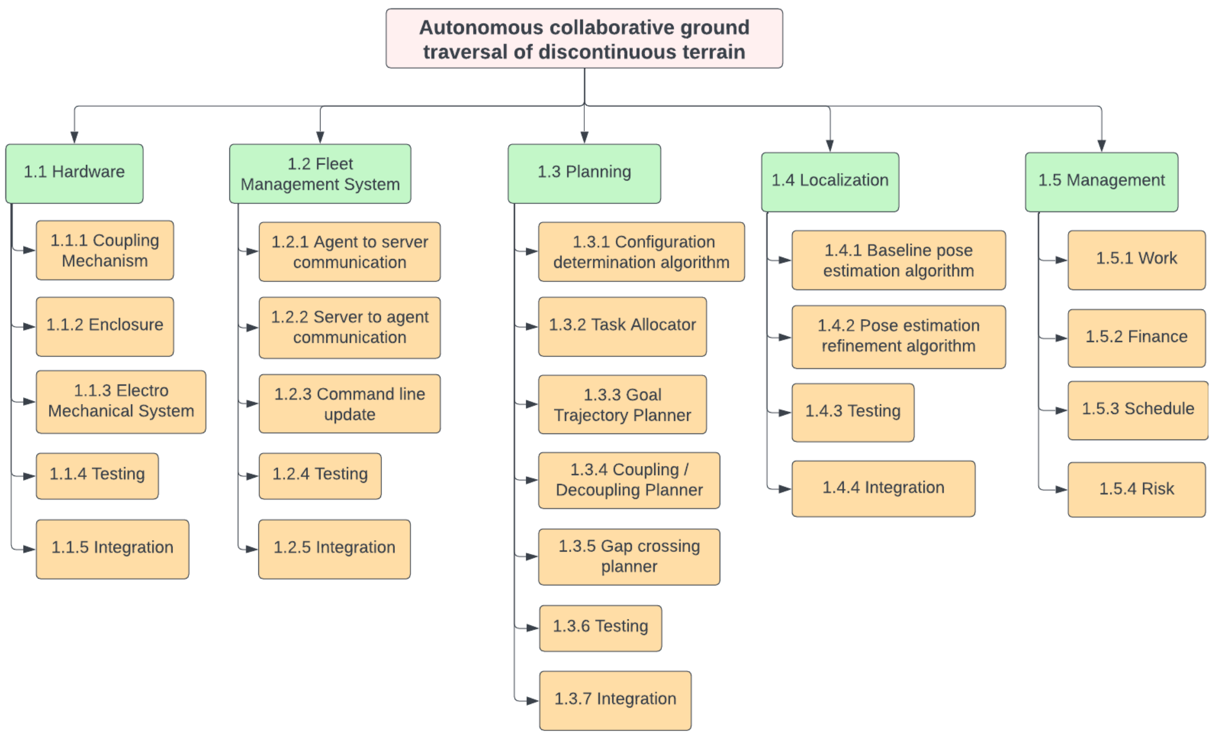


**Fig 12: WBS Schematic**

## 8.2 Schedule

Below are the subsystem-level overviews of key tasks from our work breakdown visualized vis-a-vis our internal and external milestones. We treat integration and testing as a separate entity for scheduling purposes since it would require rotation of team members to be most effective. We have used teamgantt.com to generate these visuals, and will be using Jira to track progress through the semester. The full bi-weekly breakdown can be found in the resources section

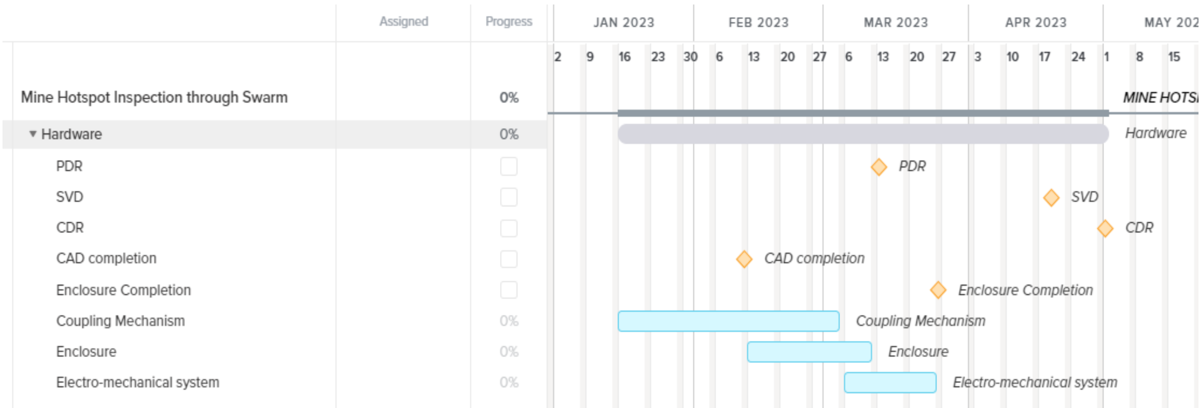


Fig 13: Hardware Schedule

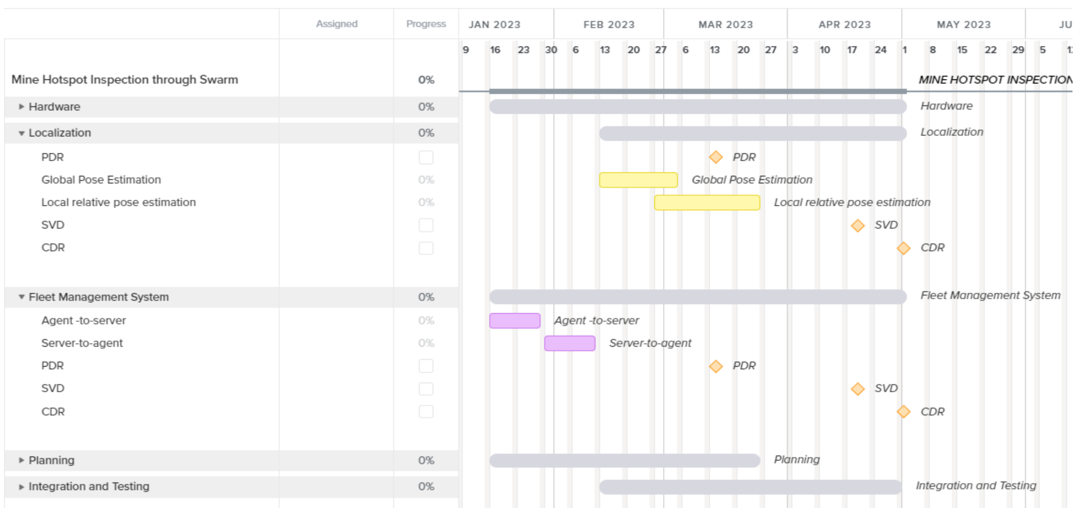


Fig 14: Localization and FMS schedule

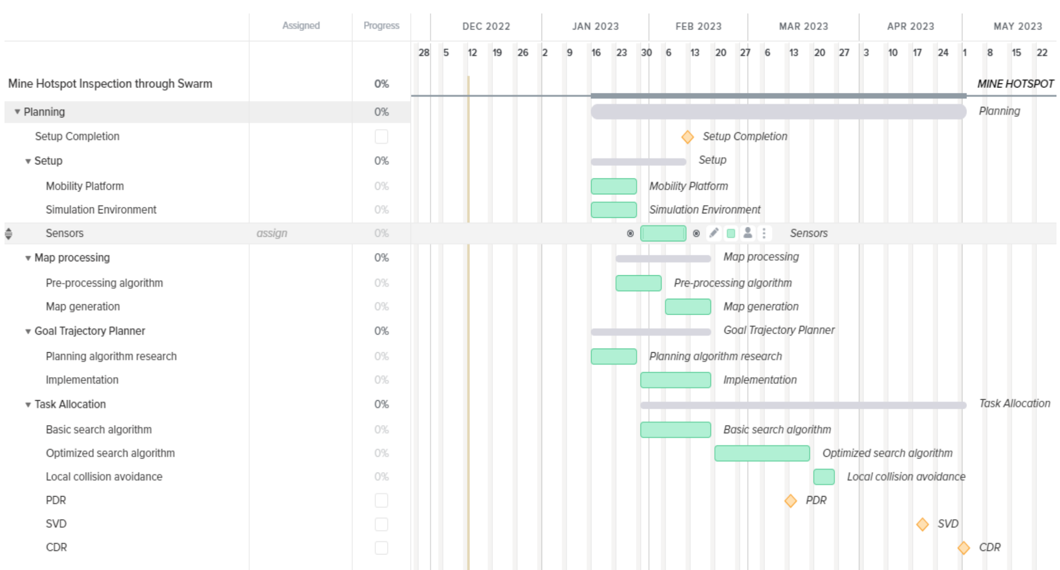


Fig 15: Planning schedule

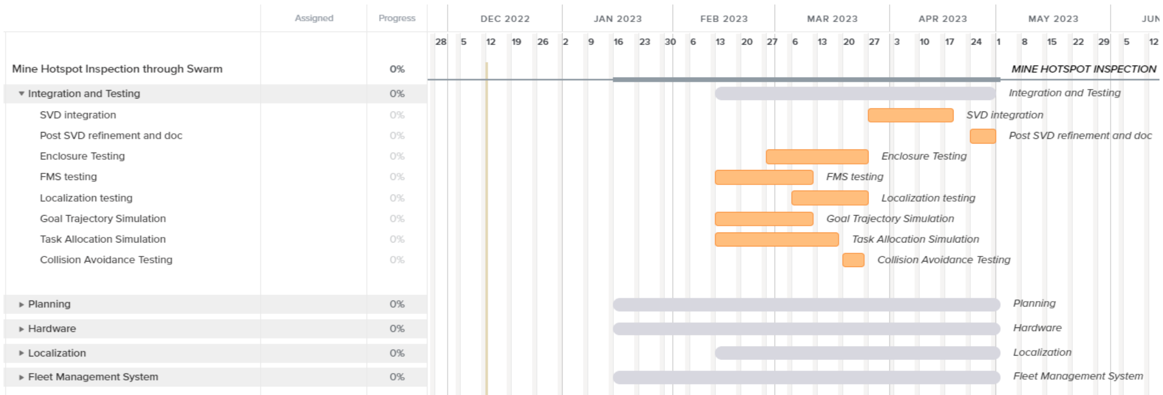


**Fig 16: Integration and Testing schedule**

## 8.3. System Validation Demonstrations

### 8.3.1 Summary of Key External Milestones

**Table 8: External Milestones Summary**

| Milestone | Date | Semester |
|---|---|---|
| Preliminary Design Review | March 13, 15 | Spring ‘23 |
| Spring Validation Demonstration | April 19 | Spring ‘23 |
| SVD Encore | April 26 | Spring ‘23 |
| Critical Design Review | May 1, 2002 | Spring ‘23 |
| System Development Review | October 15 * | Fall ‘23 |
| Fall Validation Demonstration | November 14-21 * | Fall ‘23 |
| FVD Encore | November 21-28 * | Fall ‘23 |

### 8.3.2 Progress Reviews

**Table 9: List of PR tasks**

| | |
|---|---|
| 1 | Process map of test environment |
| 2 | Implemented Fleet Management System |
| 3 | Demonstrate simulation environment |
| 4 | Demonstrate agent’s ability to execute control inputs |
| 5 | Demonstrate agent-server communication |
| 6 | Updated design of coupling mechanism |

### 8.3.3 Spring Validation Demonstration and Encore

On simulated or physical agents, we shall demonstrate the following:

**Table 10: SVD Demonstration Summary**

| | Action Item | Requirement |
|---|---|---|
| 1 | Task allocation algorithm | F2 |
| 2 | Goal trajectory planner | F6 |
| 3 | Engagement of coupling mechanism | P3 |
| 4 | Unit locomotion of two or more agents after coupling | P3 |
| 5 | Agent's ability to localize within a given map | F1 |
| 6 | Collision-free trajectory generation | F2/ P2 |
| 7 | Weight bearing capabilities of coupling mechanism | P6 |

The demonstration shall take place in the test area of the Advanced Agent-Robotics Technology Lab. The equipment shall include:

1. Khepera IV robots
2. Central compute (Laptop)
3. WiFi
4. Enclosure for Robot
    a. Controller interface
    b. Actuators
5. LIDAR attachment for Khepera

The team shall design a test environment for the above algorithms. The testing procedure will be in accordance with the use case, i.e., the robots will be given a map of the environment and placed together at a starting point.

The server will run the task allocation algorithm and send commands to each agent based on the goal trajectory planner. The agents shall demonstrate collision-free movement to the goal location.

For the coupling mechanism, the team will demonstrate tests on the load bearing capabilities of an enclosure, both in simulation and physically. The coupling mechanism will be engaged and the coupled robots will be shown moving as a single unit.

### 8.3.4 Fall Validation Demonstration and Encore

The team shall demonstrate:

**Table 11: FVD Demonstration Summary**

| | Action Item | Requirement |
|---|---|---|
| 1 | System's capability to determine feasibility of gaps in the environment | F3, P1 |
| 2 | System's capability to determine and achieve coupled configuration | F4 |
| 3 | Gap crossing by coupled system | F5, P4 |
| 4 | Alignment-insensitive coupling mechanism design | P7 |
| 5 | Formations of 3 or more robots | P3 |
| 6 | Simultaneous navigation of entire swarm | F2 |
| 7 | Integrated task allocation and trajectory tracking | F2 |
| 8 | POI exploration and visualization of sensor data | F6 |

The demonstration shall take place in the test area of the Advanced Agent-Robotics Technology Lab. The equipment shall include:

1. Khepera IV robots
2. Central compute (Laptop)
3. WiFi
4. Enclosure for Robot
    a. Controller interface
    b. Actuators
5. LIDAR attachment for Khepera

The robots will be given a map of the environment and placed at an unknown starting point. They will localize themselves within the environment and the server will produce a high-level plan. This plan will include configurations and feasibility of gaps, given the POI locations within the map.

The agents will move to their respective goal locations. When a gap is encountered, they will couple autonomously to cross it, as per the predetermined configuration. The system shall reach given POIs and cross given gaps in accordance with the performance requirements.

## 8.4 Project management: Team member responsibilities

**Table 12: Team Responsibility Breakdown**

| Team Member | Primary Responsibilities | Secondary Responsibilities |
|---|---|---|

| | | |
|---|---|---|
| Sankalp Chopkar | Hardware Development | Agent Behavior |
| Hari Shankar Ravisankaran | Agent Behavior | Fleet Management System |
| Sudhansh Yelishetty | Swarm Behavior, FMS | Agent Behavior |
| Sandhya Vidyashankar | Locomotion, Swarm Behavior | Hardware, Agent Behavior |
| Dhanvi Sreenivasan | Agent Behavior | Hardware, Locomotion |

## 8.5 Project management: Parts list and budget

**Table 13: ROM Budget Requirement**

| Component | Description | Cost ($) | Quantity | Cost of Acquisition ($) |
|---|---|---|---|---|
| Khepera IV | Agent Mobility Platform | 0 | 6 | 0 |
| YDLIDAR X4 | LIDAR | 100 | 6 | 600 |
| Actuonix Motion Devices Inc L16-100-63-12-P | Miniature Linear Actuator | 200 | 8 | 1600 |
| High Current DC Motor Drive - 43A AD-MD6321 | Motor Driver for Linear Actuator | 42.33 | 8 | 338.64 |
| Arduino Uno | Coupling Mech. Controller | 27.6 | 8 | 220.8 |
| Mini Solenoid Lock 5V1A | Latching Actuator for Pins | 12 | 16 | 192 |
| Ball Bearing 5mm | Roller Support for the Pin | 5.41 | 24 | 129.84 |
| PCB | PCB to facilitate Coupling Mechanism | 50 | 8 | 400 |
| Rail and Guide | Support for the Pin | 70 | 8 | 560 |
| Machined Parts | Housing for Coupling Mech. | 100 | 6 | 600 |
| | | **Total** | **98** | **4641.28** |

## 8.6 Risk Management

**Table 14: Key Risks and Mitigation Plans**

| Sr No | Risk Description | Likelihood | Consequence | Mitigation Plan |
|---|---|---|---|---|
| 1. | High lead time for actuators | 4 | 4 | ● Order parts by the 2nd week of Feb, with sufficient spares<br>● Identify local suppliers and alternate models as backup |
| 2. | Interference between enclosure and Khepera sensor during testing | 3 | 5 | ● Unit tests for sensor-enclosure integration<br>● Alternate material selection to avoid emf interference |
| 3. | Enclosure breaks during testing | 3 | 3 | ● Keep spare enclosures on hand<br>● Document enclosure attachment as an SOP for quick reattaching |
| 4. | FMS implementation time exceeds budgeted time | 2 | 4 | ● Review code early and identify points of failure during the winter<br>● Perform trade studies for alternate communication systems |
| 5. | Team Member(s) fall sick | 1 | 2 | ● Keep documentation of work up-to-date<br>● Distribute work such that two members are involved in any critical task<br>● Rest days to prevent burnout and stress |

# 9. References


1. Sha Yi, Zeynep Temel, and Katia Sycara. “PuzzleBots: Physical Coupling of Robot Swarm.” IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2021.
2. Sha Yi, Zeynep Temel, and Katia Sycara. “Configuration Control for Physical Coupling of Heterogeneous Robot Swarms.” IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2022.
3. Saab, W., Racioppo, P., & Ben-Tzvi, P. (2019). A review of coupling mechanism designs for modular reconfigurable robots. doi:10.1017/S0263574718001066
4. Khepera IV manual and datasheet https://www.k-team.com/khepera-iv
5. Full project schedule for spring: https://app.teamgantt.com/projects/gantt?ids=3327849