**Geomates Agent**

Intelligent Cooperative Agents

This document is project report document for an agent implementation for cooperative physical simulation game GeoMates

Project Group members

Sandesh Gavhane

[sandesh.gavhane@student.uni-luebeck.de](mailto:s%61%6e%64%65%73%68%2e%67avhan%65@s%74u%64%65n%74%2eu%6e%69-l%75e%62ec%6b%2e%64%65)

Vaibhav Sharma

[vaibhav.sharma@student.uni-luebeck.de](mailto:vai%62hav%2e%73h%61%72%6d%61@s%74u%64%65%6e%74%2euni-lue%62e%63%6b%2e%64e)

Shengyong Jiang

[sheng.jiang@student.uni-luebeck.de](mailto:%73h%65%6e%67%2e%6ai%61n%67@%73%74%75%64e%6e%74%2eu%6ei%2dlue%62%65%63%6b%2e%64%65)

Intelligent Cooperative Agents (CS4519)

Universität zu Lübeck

Date: 23 March 2025

Contents

[1.Abstract 2](#_Toc193662677)

[2. Introduction 2](#_Toc193662678)

[3. Methodology 3](#_Toc193662679)

[3.1 Research approach and design 3](#_Toc193662680)

[3.1.1 Curriculum 3](#_Toc193662681)

[3.1.2 Key lecture Concepts for Geomates 4](#_Toc193662682)

[3.1.3 Base source 5](#_Toc193662683)

[3.1.3 Setting up environment in windows 5](#_Toc193662684)

[3.1.3 how to run game 5](#_Toc193662685)

[4.Results/Findings 5](#_Toc193662686)

[Challenge 1 : To identify player type 5](#_Toc193662687)

[5. Discussion 7](#_Toc193662688)

[6. Conclusion 7](#_Toc193662689)

[7. Recommendations for Future Research 8](#_Toc193662690)

[8. References 8](#_Toc193662691)

[Figure 1- Geomates game 3](#_Toc193658488)

# 1.Abstract

* This report documents the development and implementation of an intelligent cooperative agent system for the GeoMates environment.
* The system consists of two primary agent functions that control the movement of geometric objects (a disc and a rectangle) toward a target diamond.
* The agents implement a rule-based decision-making system with potential for cooperative behaviour.
* While the current implementation provides fundamental navigation capabilities, this report analyses its performance, technical correctness, and identifies areas for optimization.

# 2. Introduction

* Task is to build an agent for cooperative physical simulation game GeoMates, a simplified clone of “geometry friends” that ran as competition at the IJCAI conference.
* Agent needs to be able to interact with the environment in order to collect diamonds and to interact with the other agent.
* University faculty will set up an ACT-R environment so we can handle the agent similar to the tutorial example of the agent.
* There will be different trials that will need different solutions. In the end we will have a challenge with new problems to be solved – there will not be completely new components but configured differently, i.e., your agent should not depend on fixed physical parameters.

A white rectangular object with red and yellow dots

AI-generated content may be incorrect.

Figure 1- Geomates game

* The task: Your agent can be either the disc (yellow) or the block (red) – each has different motion modes (disc jumps, block changes its width-to-height ratio).
* Your goal is to catch the diamonds (orange) in order to maximize the number of points you earn.
* While some of the diamonds can be reached by an agent on its own, others may require collaboration among the agents to reach them.
* Your agent receives 2 points for each diamond collected, plus one point for any diamond collected by the other agent.
* The task comprises the following challenges:
  + 1) finding out which character your agent controls and what motion capabilities it has
  + 2) understanding the level: How can it be solved? Are there different options?
  + 3) interaction with the other agent
  + 4) design of an agent: What techniques discussed in the course could be helpful?

# 3. Methodology

## 3.1 Research approach and design

### 3.1.1 Curriculum

From Intelligent cooperative agents subject following topics were analysed for the use in design

**intelligent Cooperative Agents:**

|  |  |
| --- | --- |
| **Cognitive Mechanisms and Collaboration** | **Algorithms and Architectures** |
| Autonomous agents | problem solving by searching problem spaces |
| Agents with memory and internal states | knowledge representation |
| Agents with goals und proactive agents | logics for knowledge representation |
| Perception in agents | automated reasoning |
| Embodied agents | knowledge bases and applied reasoning |
| Agents interacting with real environments | inductive logic programming |
| Control of dynamics and belief updating | Knowledge Graph Embeddings |
| Learning through interaction | Introduction to Multi-Agent Systems (MAS) |
| Sense of control and minimal active self | Negotiation and Multi-Agent Systems |
| Cooperative agents and cooperative AI | Formal Argumentation in Multi-Agent Systems |
| Theory of mind and anticipation | foundations of AI planning |
| Dynamic decision making | PDDL and HTN planning (lecture slides) |

### 3.1.2 Key lecture Concepts for Geomates

For GeoMates, following key concepts can be directly applied:

1. **Problem Solving by Searching Problem Spaces**
   * Path planning for agents
   * Finding optimal routes to diamonds
   * Collision avoidance strategies
2. **Knowledge Representation & Logic**
   * Representing game state
   * Modelling physical constraints
   * Encoding agent capabilities
3. **Multi-Agent Systems (MAS)**
   * Disc and rectangle agent cooperation
   * Shared goal achievement
   * Resource coordination
4. **PDDL and Planning**
   * Action planning for diamond collection
   * Sequencing movements
   * Coordinating joint actions

### 3.1.3 Base source

* Base source code for Geomates environment by cloning repository

https://gitlab.isp.uni luebeck.de/hai/geomates

* The Readme document explains how to install the software
* Faculty provided source code and a docker file for virtualization.
* There was a dummy Act-R agent as starting point.
* It was allowed to include external components as well (e.g., a PDDL planning system, theorem prover, etc.).
* In the agent code, there is an example of how regular code or external software can be connected.

### 3.1.3 Setting up environment in windows

Please find on the following location, how to setup environment in windows for Geomates game

<https://github.com/SandeshGavhane/DeepSeek_ICA_Agent/tree/main/docs>

### 3.1.3 how to run game

Please find on the following location, how to run Geomates game

<https://github.com/SandeshGavhane/DeepSeek_ICA_Agent/tree/main/docs>

# 4.Results/Findings

## 4.1 Challenge 1 : To identify player type

Agent model submitted on 18 March 2025 do following steps to identify the player type

After warming up and sending random keys , ‘a’ key will be pressed and rect player position will be checked and if it changed then it will be RECT player otherwise DISC player.

But this will cause problem when two agents starts playing and do the same actions and there will be confusion

Solution : solution to this problem was found by updating act-r-experiment.lisp

Read-player-type function implemented which will check (:playing rect) or (:playing disc) message received from Game server.there was challenge as initially some telnet codes were sent and was not easy to implement this.

A screenshot of a computer program

AI-generated content may be incorrect.

A screenshot of a computer code

AI-generated content may be incorrect.

Player info will be continuously update in visicon so that in model we can read it

A screenshot of a computer program

AI-generated content may be incorrect.

Working code for this can be found on the following location

<https://github.com/SandeshGavhane/DeepSeek_ICA_Agent/blob/main/geomates/DeepSeekAgent_Version3_Reactive_PlayerDetection.lisp>

<https://github.com/SandeshGavhane/DeepSeek_ICA_Agent/blob/main/geomates/act-r-experiment.lisp>

## 4.2 Video of Agent

Following agent is used for video

<https://github.com/SandeshGavhane/DeepSeek_ICA_Agent/blob/main/geomates/DeepSeekAgent_Version2.lisp>

<https://github.com/SandeshGavhane/DeepSeek_ICA_Agent/blob/main/geomates/navigation-functions.lisp>

Please find on the following location, video AgentRunningVideo.mp4

<https://github.com/SandeshGavhane/DeepSeek_ICA_Agent/tree/main/docs>

## 4.3 Experimental Results

### 4.3.1 Navigation Test Scenarios

A series of test scenarios were conducted to evaluate agent performance:

| **Test Case** | **Starting Positions** | **Target Position** | **Expected Behavior** | **Actual Behavior** | **Pass/Fail** |
| --- | --- | --- | --- | --- | --- |
| 1 | Disc: (0,0), Rect: (2,2) | Diamond: (5,5) | Both move toward target | Disc moves up-right, Rect moves right-up | Pass |
| 2 | Disc: (10,5), Rect: (0,0) | Diamond: (5,5) | Disc moves left, Rect moves right | Disc prioritizes up movement | Fail |
| 3 | Disc: (5,5), Rect: (5,5) | Diamond: (5,5) | Both agents stop | Both return nil | Pass |

### 4.3.2 Performance Analysis

The current implementation shows mixed results:

* **Successful Convergence**: Both agents successfully reach the target when starting from certain positions
* **Path Inefficiency**: The disc agent takes non-optimal paths due to its movement priority logic
* **Incomplete Vertical Logic**: The rectangle agent cannot navigate downward, limiting its movement range

### 4.3.3 Cooperative Behaviour Testing

Tests for potential cooperative behaviour were inconclusive as the communication channel between agents (the rectangle parameters passed to the disc function) is not utilized in the current implementation.

# 5. Conclusion

* The GeoMates navigation system demonstrates a basic implementation of intelligent agents with rule-based decision-making.
* While the current implementation successfully navigates toward targets in simple scenarios, it lacks the sophistication for optimal pathfinding and true cooperative behaviour.
* The project shows promise in its approach to geometric agents with different physical properties (disc vs. rectangle) and its foundation for potential cooperation.
* With the recommended improvements, particularly in completing the vertical movement logic and implementing genuine cooperative behaviour, the system could evolve into a robust navigation framework for multiple agents in complex environments.

# 6. Recommendations for Future Research

|  |  |
| --- | --- |
| **Problem Solving by Searching Problem Spaces** | **Current status** |
| 1. Path planning for agents | implemented |
| 1. Finding optimal routes to diamonds | implemented |
| 1. Collision avoidance strategies | implemented |
| **Knowledge Representation & Logic** |  |
| * Representing game state | Future work |
| * Modelling physical constraints | Future work |
| * Encoding agent capabilities | Future work |
| **Multi-Agent Systems (MAS)** |  |
| * Disc and rectangle agent cooperation | Future work |
| * Shared goal achievement | Future work |
| * Resource coordination | Future work |
| **PDDL and Planning** |  |
| * Action planning for diamond collection | Future work |
| * Sequencing movements | Future work |
| * Coordinating joint actions | Future work |

# 7. References

* Source code is maintained on the GitHub and can be found on the following location

[SandeshGavhane/DeepSeek\_ICA\_Agent: ICA Agent for Geo Mates game](https://github.com/SandeshGavhane/DeepSeek_ICA_Agent)

* Model which was submitted on the 18 March 2025 challenge can be found on the following location

<https://github.com/SandeshGavhane/DeepSeek_ICA_Agent/blob/main/geomates/DeepSeekAgent_Version1.lisp>

* There will be continuous update for the project and latest code can be found on [SandeshGavhane/DeepSeek\_ICA\_Agent: ICA Agent for Geo Mates game](https://github.com/SandeshGavhane/DeepSeek_ICA_Agent)
* code files will also be submitted with project report