

Multi-Agent Programming Contest 2017

Participation Registration

Jonathan Pieper

Department of Informatics, Clausthal University of Technology

Introduction

1. What is the name of your team?

Busy Beaver

2. Who are the members of your team? Please provide names, academic degrees and institutions.

Jonathan Pieper, B.Sc. Student at Clausthal University of Technology, jpieper@swbmail.de

3. Who is the main-contact? Please also provide an Email address.

Please direct any communication to each nonempty element of the power set of team members.

4. How much time (developer hours) will you have invested (approximately) until the tournament?

Approximately 200 hours

System Analysis and Design

1. Briefly, what is the **main strategy** of the team?

When pursuing a job, a group of agents will form, buy base items, meet at a workshop and subsequently assemble and deliver the required items. Idle agents may collect resources, proactively buy items or irritate the opposing team by driving into a one-way dead end.

2. Will you use any existing multi-agent system **methodology** such as Prometheus, O-MaSE, or Tropos?

No

3. Do you plan to **distribute** your agents on several machines?

Communication between agents will only take place via the .send and .broadcast actions, enabling distribution over several machines. However, in order to keep the agents from feeling lonely and to ensure optimal (i.e. no) latency during communication, the entire team is planned to run on a single machine. Additionally, for the sake of simplicity and reduced memory usage, some static data, such as the map graph, will be stored in one place and accessed by all agents.

4. Is your solution based on the **centralisation** of coordination/information on a specific agent? Conversely if you plan a decentralised solution, which strategy do you plan to use?

There is no specific centralised agent. Since the team will have both a communication strategy and a coordination strategy, a decentralised solution strategy is not deemed necessary.

5. Describe the **communication strategy** in the agent team. Can you estimate the communication complexity of your approach?

The agents will periodically rotate a token. Only its holder may issue a call for proposals. This ensures (binding) proposals to be only made one at a time. The communication complexity is bounded from above by this team's namesake.

6. Describe the team **coordination strategy** (if any).

An agent holding the token can start a series of Contract Net Protocol iterations (one for each tool and each base item), in order to find help for completing a job.

7. How are the following agent features implemented: *autonomy*, *proactiveness*, *reactiveness*?

No agent will be controlled by a central entity or another agent. However, agents will offer their teammates to buy base items or tools (restricted by carrying capacity and usable tools) and plan a route through shops in advance. Unexpected problems may be solved by if-clauses or recovery plans.

Software Architecture

1. Which **programming language** do you plan to use to implement the multi-agent system? (e.g. 2APL, Jason, Jadex, JIAC, Goal, Java, C++, ...)

The agents will be programmed in AgentSpeak and Python.

2. Which **development** platform and tools are you planning to use?

Mainly Notepad++

3. Which **runtime** platform and tools are you planning to use? (e.g. Jade, AgentScape, simply Java, ...)

Windows 10 (64 bit), Python 3.5, pyson

4. Which **algorithms** will be used?

Algorithms used will include Token Ring, Contract Net Protocol and A*