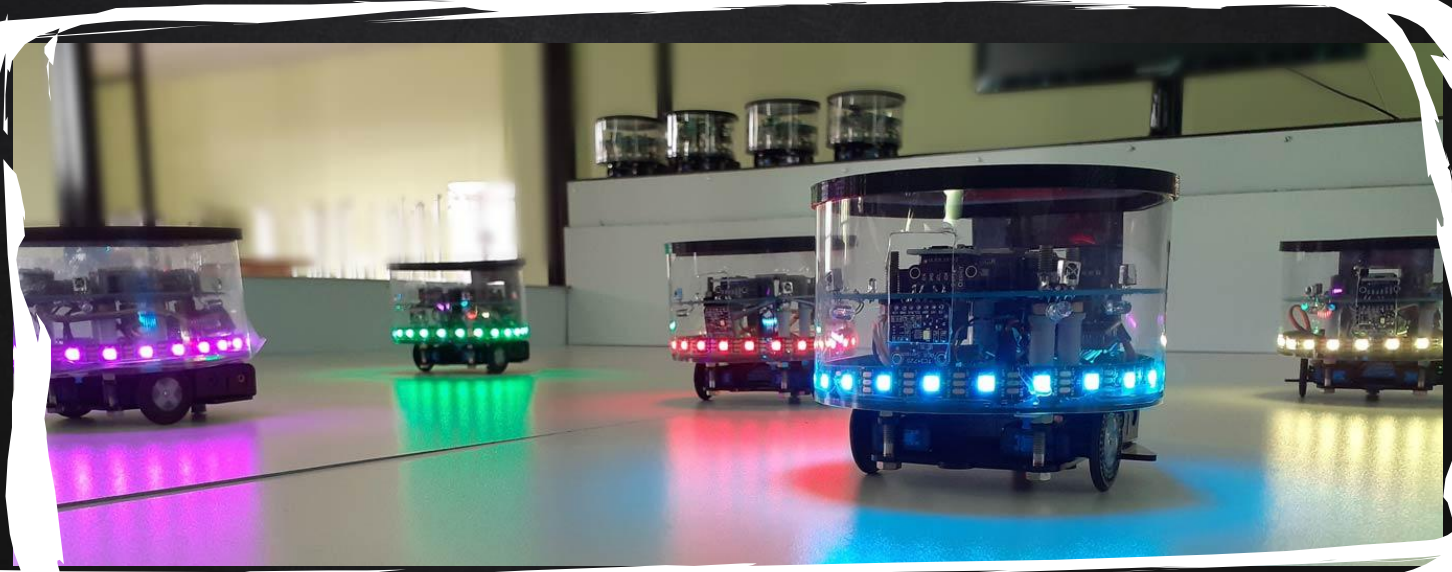
Swarm Intelligence Programming Framework



Group 03

Team members

Ekanayaka S.M.	E/16/094
Madushanka H.M.K.	E/16/221
Perera A.L.H.E.	E/16/275

Supervisors

Prof. Roshan Ragel

DR. Isuru Nawinne

Dr. Mahanama Wickramasinghe

Dr. Sithumini Ekanayake

Introduction

Introduction

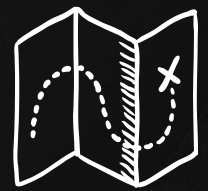
PeraSwarm is a robotic multi-agent systems testbed with different types of robot hardware, virtual robots and a simulation environment.

This project will focus on building a high-level programming and compiler framework which will support multiple different types of robot hardware (real and virtual).

Motivation

Motivation

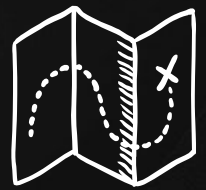
- Robot having same functionalities but with different hardware architectures are not compatible with the existing swarms.
- Setting up each of the robot agent for swarm researches is difficult due to individual attention to each of the robot.
- Make swarm robot systems ready for experiments in a small amount of time.
- Implementing a visual programming interface



Methodology

Methodology

1. Implement the remote compilation
2. Testing algorithms implemented based on FSMs.
3. Implementing complex behaviours by aggregating several FSMs.
4. Code generation and upload code to each robot.



Milestones

Milestones

1. Remote Cross Compiler Development
2. Algorithms development and testing
3. Finite State Machines (FSMs) Testing
4. Aggregator FSM Testing
5. Code Generation and Uploading to Robots
6. Testing SAR (Search and Rescue) on Arena

Analysis

Existing Frameworks

Framework	Orchestration	Type of Robots	Testing Method
Buzz	Decentralized	Heterogeneous	Simulations
ROS	Centralized	Heterogeneous	Real
Karma	Centralized	MAVs	Simulations
PaROS	Centralized	UAVs	Real / Simulations

Existing Frameworks (Continue)

Framework	Orchestration	Type of Robots	Testing Method
CrazySwarm	Centralized	Homogenous	Real
EmsBot	Centralized	Heterogeneous	Simulations
EmsBot Script	Centralized	Heterogeneous	Simulations

Key takeaways from framework studies

1. Actor based Programming Frameworks
2. Domain Specific Languages
3. Frameworks based on CBSD
4. Anti-Ros Pattern based approaches
5. UI based codeless programming approach
6. Virtual Machine based programming frameworks

1. Actor Based Programming Frameworks

- Decouple high level task programming instead individual robot programming
- Intricate only task coordination
- Different plugins defines different robots

Key takeaways from framework studies

1. Actor based Programming Frameworks
2. Domain Specific Languages
3. Frameworks based on CBSD
4. Anti-Ros Pattern based approaches
5. UI based codeless programming approach
6. Virtual Machine based programming frameworks

2. DSL (Domain Specific Languages)

- Ex : Buzz, ROS, Karma, PaROS
- Neighbouring operations, virtual stimergy, collective decision making
- PaROS
  - Abstract Swarm as the Actor
  - Support Drone Enumeration,
  - Task partitioning, Path planning, fault detection, correction

Key takeaways from framework studies

1. Actor based Programming Frameworks
2. Domain Specific Languages
3. Frameworks based on CBSD
4. Anti-Ros Pattern based approaches
5. UI based codeless programming interface
6. Virtual Machine based programming frameworks

3. Frameworks based on CBSD (Component Based Software Development)

- Used component based approaches.
- GSDF – Generic Development Framework for Swarms
  - Compatible with ROS
- EmSBot Modular based Framework
  - Consider resource constraints
  - Real time support

Key takeaways from framework studies

1. Actor based Programming Frameworks
2. Domain Specific Languages
3. Frameworks based on CBSD
4. Anti-ROS Pattern based approaches
5. UI based codeless programming approach
6. Virtual Machine based programming frameworks

4. Anti-Ros Pattern based approaches

- Python based robot programming framework – CrazySwarm
- Support only homogenous robots swarms
- Optimized and Low latency communication

Key takeaways from framework studies

1. Actor based Programming Frameworks
2. Domain Specific Languages
3. Frameworks based on CBSD
4. Anti-Ros Pattern based approaches
5. UI based codeless programming approach
6. Virtual Machine based programming frameworks

5. UI based codeless programming interface

- Reactile Project – Physical manipulations to swarms
- Used UI programming practices.
- Less programming required for users.

Key takeaways from framework studies

1. Actor based Programming Frameworks
2. Domain Specific Languages
3. Frameworks based on CBSD
4. Anti-Ros Pattern based approaches
5. UI based codeless programming approach
6. Virtual Machine based programming frameworks

6. Virtual Machine based programming frameworks

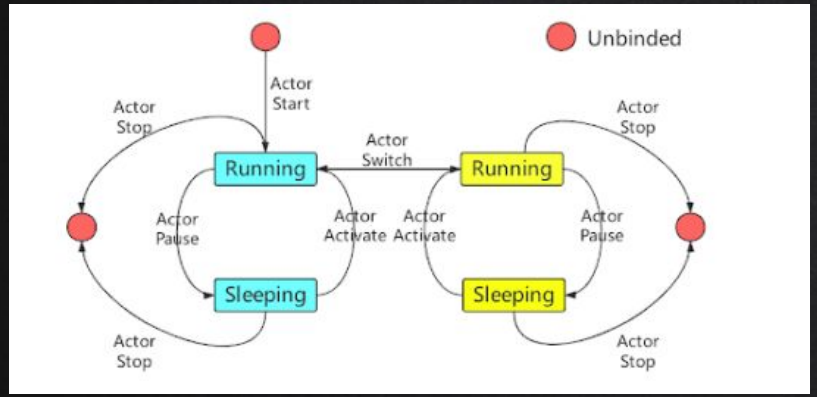
- EmSBot Scripts framework
- Solve heterogeneity problem
- Address implementation challenges
  - Concurrency – using codelet functions
  - Reduce memory consumption – using virtual pagination.

Visual Programming Languages

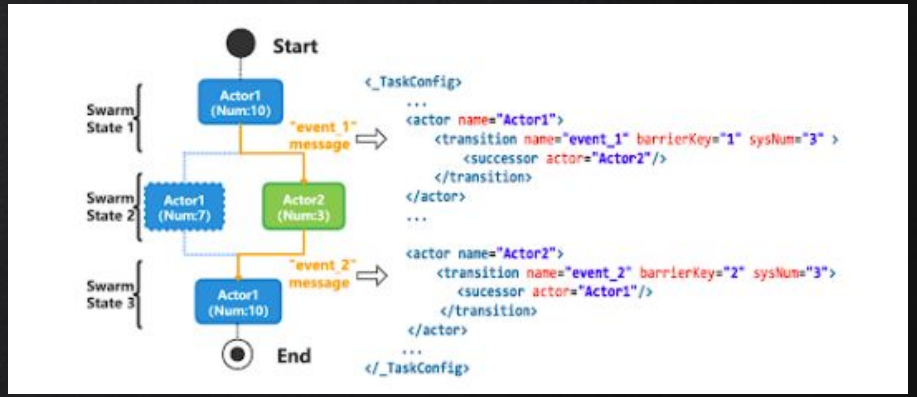
- Visualize programming logic achieve simplicity..
- Less burden handling syntax errors.
- Existing VPLs
  - Smart Blocks – Event Condition Action
  - Flow Board – using visual flows

Swarm Behavior Implementations

- Bottom up behavior modeling approaches are considered.
- Behavior programming – State machines
- Pheromone Communication – Flow chart
- Provide freedom to introduce new behaviors.



State Machine – Behavior

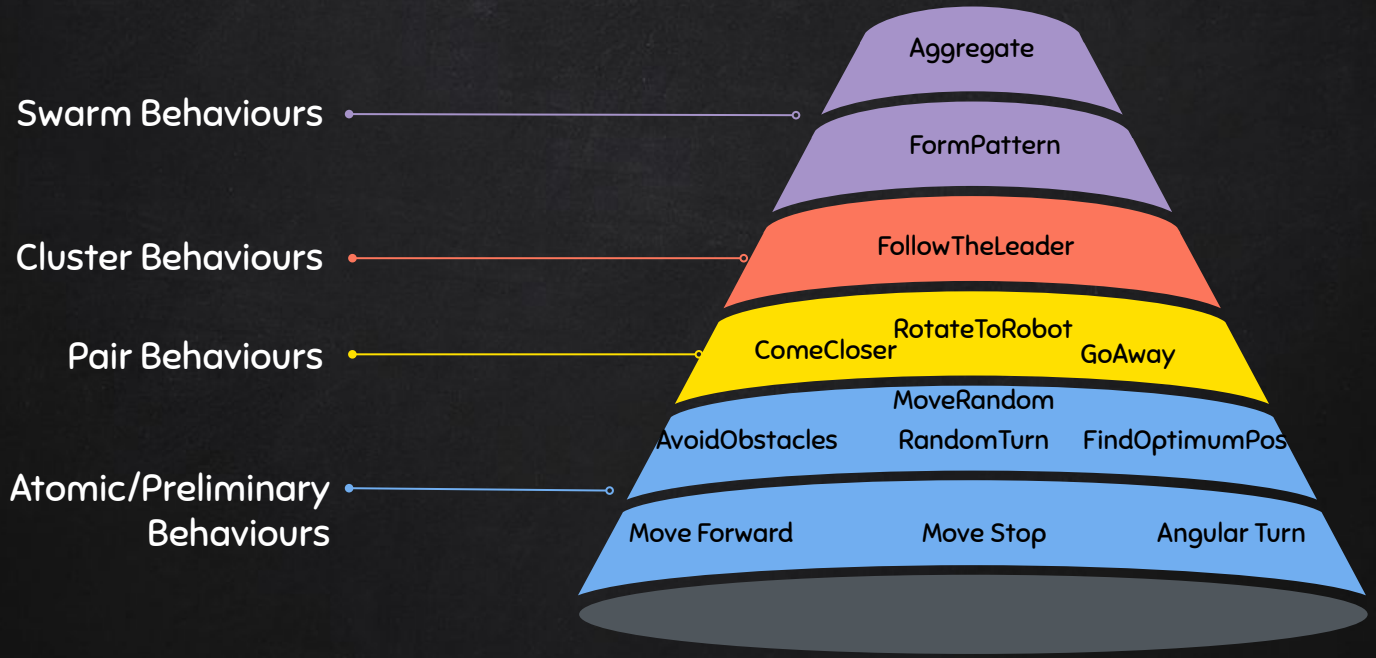


FlowChart – Change behavior

Work on the project so far

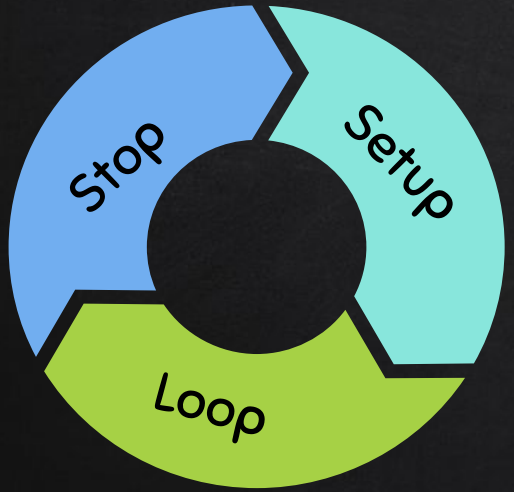
The Big Picture

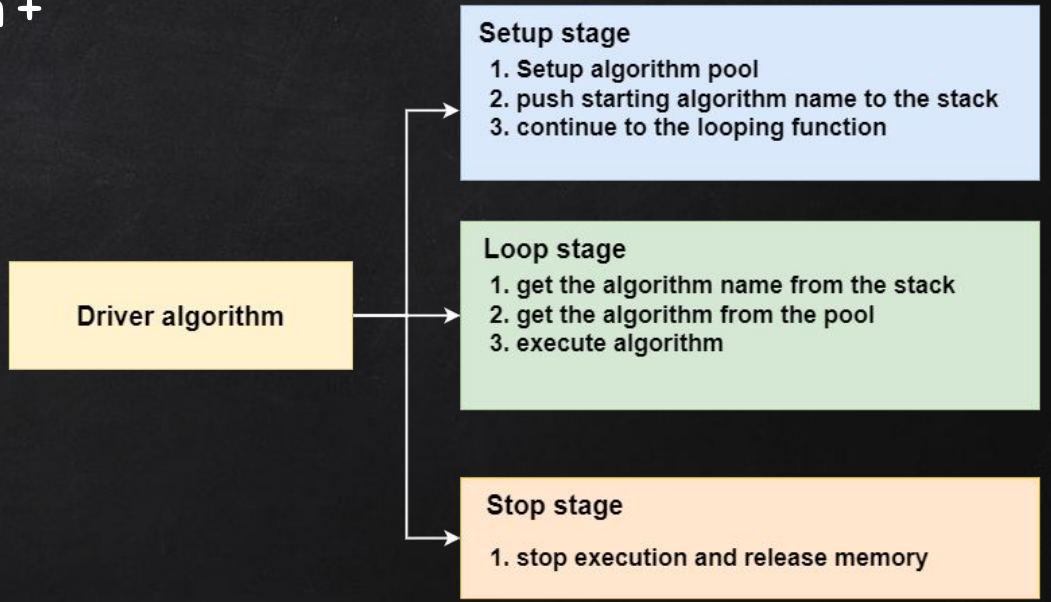
Bottom-up Approach of swarm behaviours



Switching Between Behaviours?

✘ Using a driver algorithm + Execution Stack





Research Questions and Gaps

- How new robots are introduced into the swarm having same functionalities but different hardware architectures ?
- Introduce virtual robots into the swarm.
- Configure the robots over the air ?

Conclusion

- Framework should be able to abstract the underlying hardware architectures.
- More behaviors integration provides more benefits for research purposes.
- Providing VPLs enhance the simplicity and usability of the programming framework.

Thank you!