Final Proposal

Team 16

Abstract/Problem

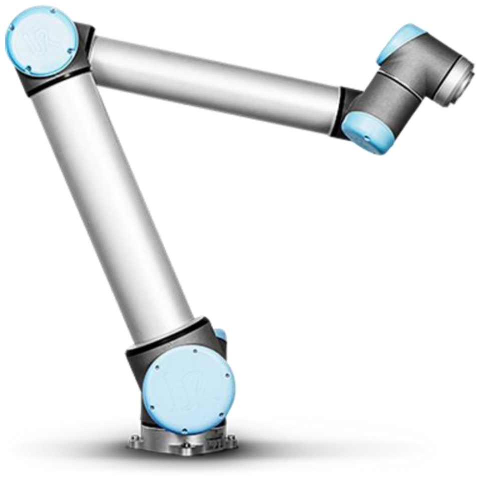
- Today's factories are limited by the specialization of the machines in them.
- To fix this a software is needed to allow a more generalization of factories.
- Once the solution is found, multiple different robots will be able to cooperatively complete a task.
- These robots need to communicate effectively, but all have different native languages

Objective

- Cooperative 3D Printing with Industrial Robotic Arms with ROS
- Developing a universal printing interface
- Design/Implement a communication protocol for 3rd Party robotic arms

Key Concepts

- Swarm Manufacturing
- 3D Printers
- ROS
- MoveIt
- Gazebo
- Degrees Of Freedom
- Linux
- Computer Aided Design (CAD)
- G-Code
- Slicer
- Python
- UR10
- Kinova Gen3



UR10

Kinova

Related Work

- Additive Manufacturing
  - 3D System's SLA-750
- Swarm Intelligence
  - Unbox Robotics
- Rosotics
  - Rapid Induction Printing

Requirements

- Ability to input a G-Code file to the machine and have two different robots 3D-Print the object in the G-Code file
- Code should be generic and be able to work on multiple different brands of robotic arm
- Potentially add a slicer so that an object from a CAD software can be directly inputted without the prior translation to G-Code (this is a "stretch goal")

Use Cases

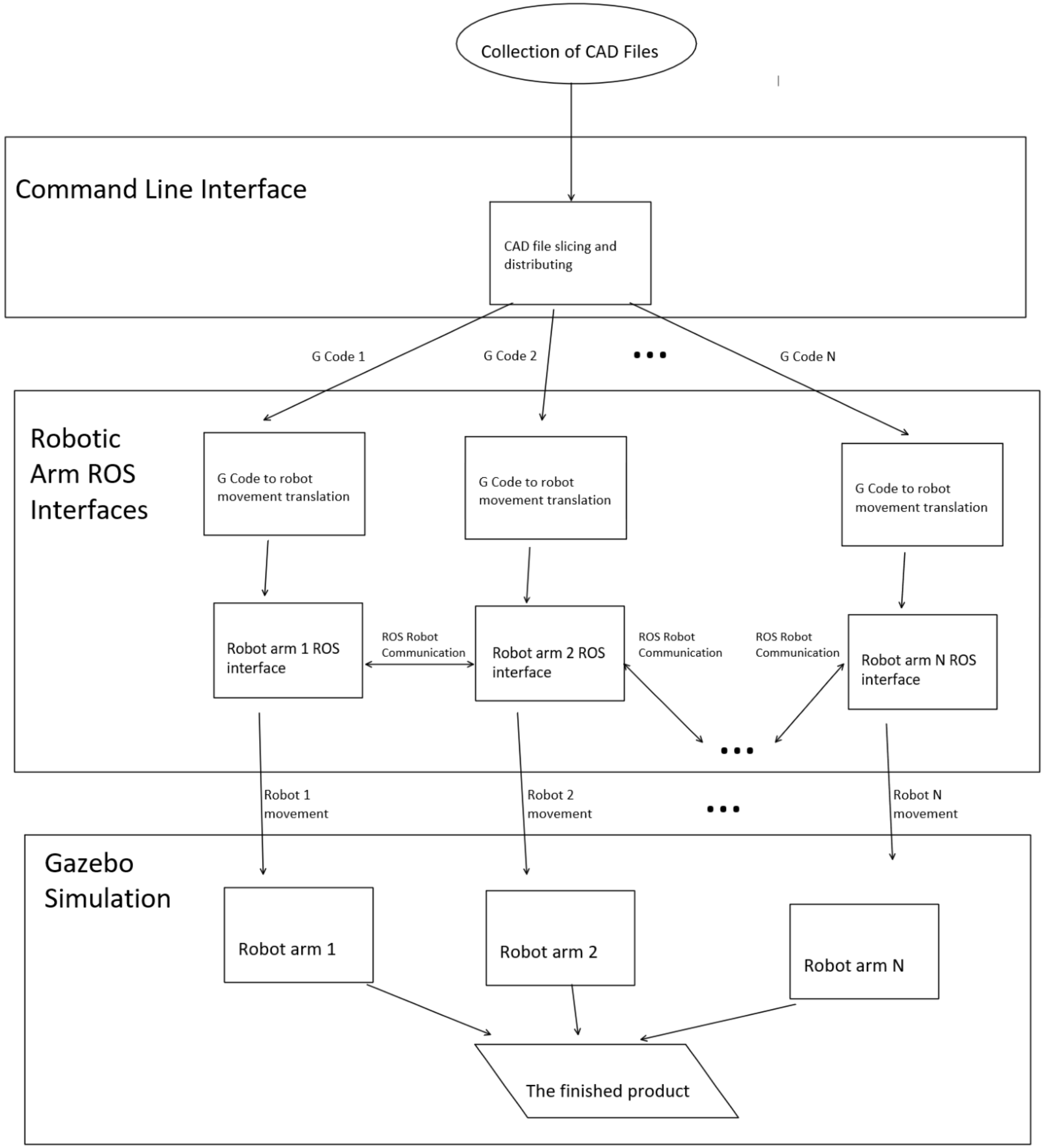
Current:

- AMBOTS Mechanical/Software Engineers
  - Knowledge of command line interface along with environment setup with ROS, MoveIt, Robotics Arms, etc is required.

Potential:

- Construction companies will use this to build housing with cement.
- General manufacturing companies can use this technology to improve their factories efficiency's.
- Any 3d printing job that is difficult to produce with a single printer, such as large objects, multipart objects, etc.

High Level Architecture



Tasks/Timeline

Tasks	Dates
1. Research ROS, MoveIt, and Gazebo	11/07 - 11/14
2. Research Robotic arm options and select two arms of different brands.	11/14 - 11/21
3. Document set up of ROS, MoveIt, the robotic arms, and the Ubuntu environment	11/21 - 12/05
4. Research how each arm receives commands.	12/05 - 12/12
5. Simulate basic movement of robotic arm choices in gazebo	12/12 - 12/26
6. Research G-Code	12/26 - 01/02
7. Download and use slicer to create G-Code file.	01/02 - 01/09
8. Pipe G-Code files to the first robotic arm by creating a program to translate from a G-Code file to robotic arms' native language	01/09 - 01/23

Tasks/Timeline Cont.

9. Pipe G-Code files to the second robotic arm by creating a program to translate from a G-Code file to robotic arms' native language	01/23 - 02/06
10. Simulate the G-Code movement of robotic arm choices in gazebo using basic G-Code files.	02/06 - 02/13
11. Find and setup a concrete or material extruder for each robotic arm.	02/13 - 02/20
12. Simulate printing using the extruder in gazebo.	02/20 - 03/06
13. Run both robotic arms in the same environment.	03/06 - 03/20
14. Use ROS to communicate between the two robotic arms.	03/20 - 04/03
15. Simulate simultaneous printing with the robotic arm choices in gazebo using ROS communication.	04/03 - 04/24

Deliverables

- Research – collection of all throughout implementation process
- Documentation - how the code works, and why choices were made
- Software Package – code written to reach goal of simultaneous robotic arm 3D printing
- Tutorial – setup of programs used as well as software package produced
- Final Zip File – Includes final report along with code written

Thank You