University of Arkansas – CSCE Department
Capstone I – Preliminary Proposal – Fall 2022

AMBOTS

Cassandra Nelson, Michael Darden, Stanley Van, Alvaro Becares Fernandez

Abstract

The manufacturing industry currently uses specialized machines and factories to produce products. This limits the flexibility of what can be produced out of one factory. New machines are often required for new products. AMBOTS is using existing or developing new technologies to break down manufacturing tasks into smaller and simpler ones. These can then be automated and coordinated by robots. However, these robots need a way to communicate with one another. Our goal is to design and implement a communication protocol for different third-party robotic arms to communicate with each other to cooperatively complete a task.

We will solve this problem by breaking it down into manageable steps. First, we will select a group of robotic arms to start with and learn how to operate these arms in ROS, Moveit, and Gazebo. After this, we will create a program that can translate G-Code files into the robotic arms' native languages. Along with the translation program, another will be needed to allow the separate robots to communicate with each other. Finally, we will use both of the previous programs to perform a cooperative print using two separate robotic arms. When this goal is achieved, the third-party robots will be able to communicate with each other, perform a task cooperatively, and complete a 3D print of a file they are given.

1.0 Problem

When a new product is designed, there is a long and expensive production process. If one were to create a prototype of a car and work out all of the design flaws, one then must deal with the cost and issues of finding a way to produce the car on a massive scale. The answer to that problem in today's world is to design a factory around making that car. For several components of the car, specific machines have to be designed and built solely for that one car. This is an expensive and non-reusable solution that has become a massive challenge for almost all mass-produced products today. This also causes issues for some companies in the supply chain. It discriminates against the small guys. Because companies are so specialized on specific things, they have to rely on the services of others who may not want to provide their services as it may seem unprofitable to serve them. As Dr. Zhou put it, our civilization is built on manufacturing. Any production capability we lose is bad for our civilization. We cannot even reproduce the pyramids. We went to the moon and still have not been back.

AMBOTS

Factories are built for the product. When demand for a product decreases, we lose the factory and the capability to produce the product. We cannot reproduce the sophisticated production process we have here on Earth on Mars. These problems lead to the overarching goal of AMBOTS being to create a general-purpose factory. A large obstacle to this goal is stationery and specific machines, and AMBOTS is the pioneer of the answer. In order to rid a factory of its need for stationery and specific machinery, the production process is broken down to the assignments of specific tasks to different robots. When production changes are needed, the robots can be assigned different tasks and can move around accordingly to their new tasks. This means that there is a need for an open ecosystem software package that can give instructions to these kinds of robots and also support a wide variety of third-party robots. It would be impractical to create a whole new network of robots for this, so instead, accommodating existing robots in the industry is a more feasible and economical goal.

2.0 Objective

The objective of this project is to perform cooperative 3D printing with Industrial Robotic Arms with ROS. Developing a universal printing interface such that it is possible to control different third-party robots by integrating them into our sponsors platform and cooperate with other robots for manufacturing. We will do this by designing and implementing a communication protocol such that other third-party ROS-compatible robots can effectively talk with our robots over a local wireless network.

3.0 Background

3.1 Key Concepts

Swarm Manufacturing is a new form of manufacturing developed for future factories. It is the employment of a swarm of different robots to manufacture products cooperatively on an open factory floor.

The Universal Robots UR10 is the largest robot in the Universal Robots collaborative series. It has a payload up to 10 kg. It is very easy to set up, and it offers one of the fastest payback times in the industry. It is ROS compatible and can be simulated in both MoveIt and Gazebo.

The Kinova Gen3 Six Degrees of Freedom (6DOF) robotic arm is a 6-axis robotic arm. It is ROS compatible and can be simulated in both MoveIt and Gazebo.

The Robot Operating System (ROS) is a set of software libraries and tools that help researchers and developers build and reuse code between robotic applications. ROS 2 has support for real time code and embedded systems. Any code file that utilizes ROS is called a node. Nodes have three ways of communicating. The first way is the publisher subscriber method. The second way is through services. The third way is through actions.

MoveIt is an open-source ROS package. Its basic task is to provide the necessary trajectories for our robotic arms. This allows the robotic arms to move to the right locations. There are two main functions which are creating a plan and sending a plan. Known obstacles can be added so that they can be avoided.

Gazebo is an open-source 3D robotics simulator. It helps developers rapidly test algorithms and design robots in digital environments. This reduces the costs and safety risks of testing significantly. It can also simulate sensors. Sensors can use noise models for noise properties in the data the sensors return.

Degrees of Freedom (DOF) is the number of independent variables that define the possible positions or motions of a mechanical system in space. The number of degrees of freedom is equal to the total number of independent displacements or aspects of motion (translational or rotational).

Linux is an open-source operating system. It comes in many different distributions. The specific distribution we will be using Ubuntu 20.04. This is to allow the use of ROS 1 in case it is needed.

4.0 Approach/Design

4.1 Requirements

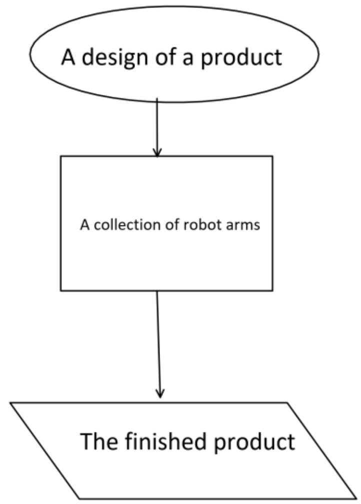
- Ability to input a G-Code file to the machine and have two different robots 3D print the object that is represented by the G-Code file.
- This code should be generalized and be able to work on multiple different brands of robotic arm. We are looking for universality.
- Potentially add a slicer so that an object from CAD software can be directly inputted without the prior translation to G-Code. This is a stretch goal.

4.2 High Level Architecture

To begin the design process, we first produced a simple high-level design. This is so that we can build more complicated modules and pipelines later on.

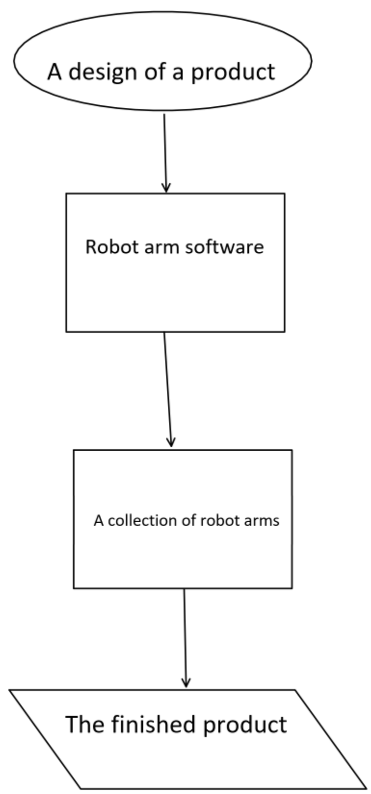
AMBOTS

Design Iteration 1:



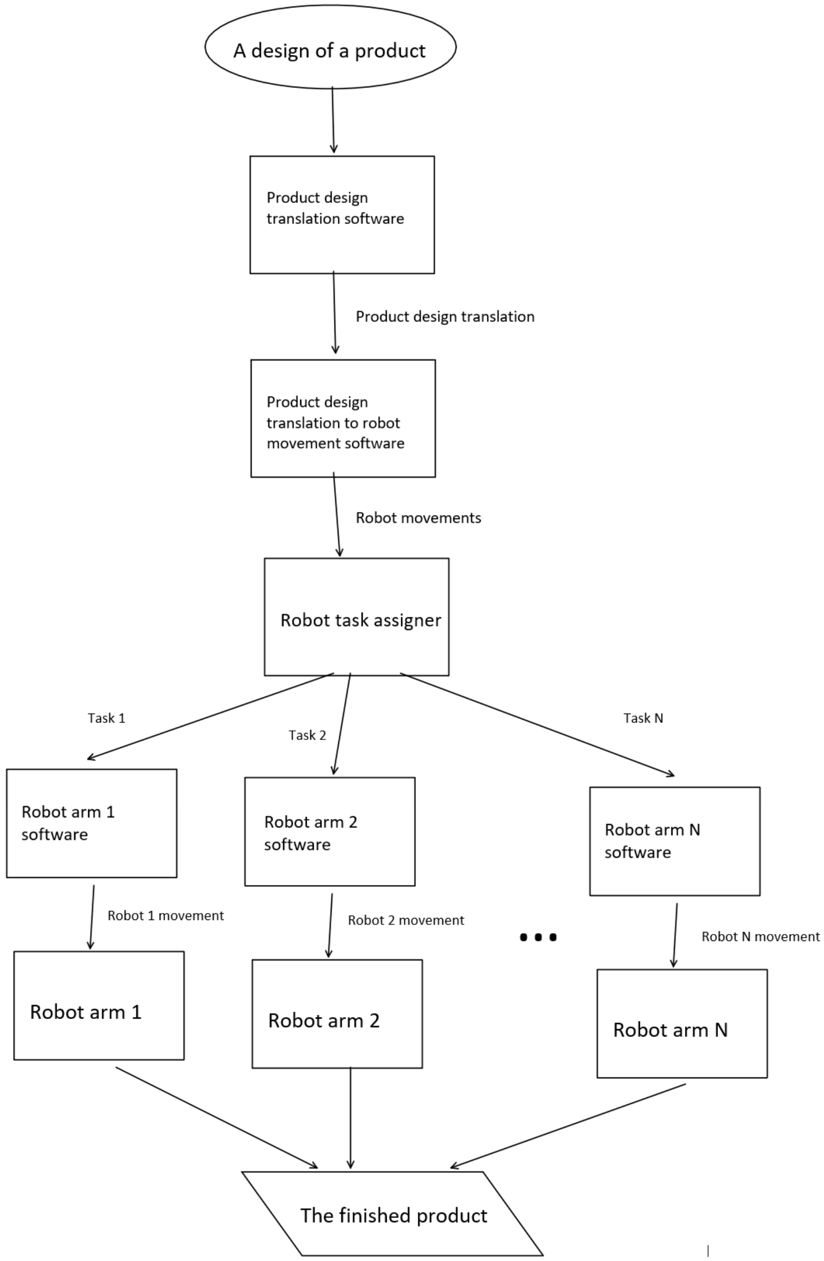
This first iteration has an input of a product design and an output of a finished product. A collection of robot arms will carry out this task. This is the highest-level description of the task, no software yet, only supplying a product design to some robots and then outputting a completed product.

Design Iteration 2:



This second iteration of the design adds a software component. This software component will receive the product design and interpret it. It handles the reading of the design, the translation of the information into a language the robots can understand, and the management of the robots to produce the product. The collection of robot arms listens to the software and follow its instructions and actually perform the tasks to produce the finished product.

Design Iteration 3:

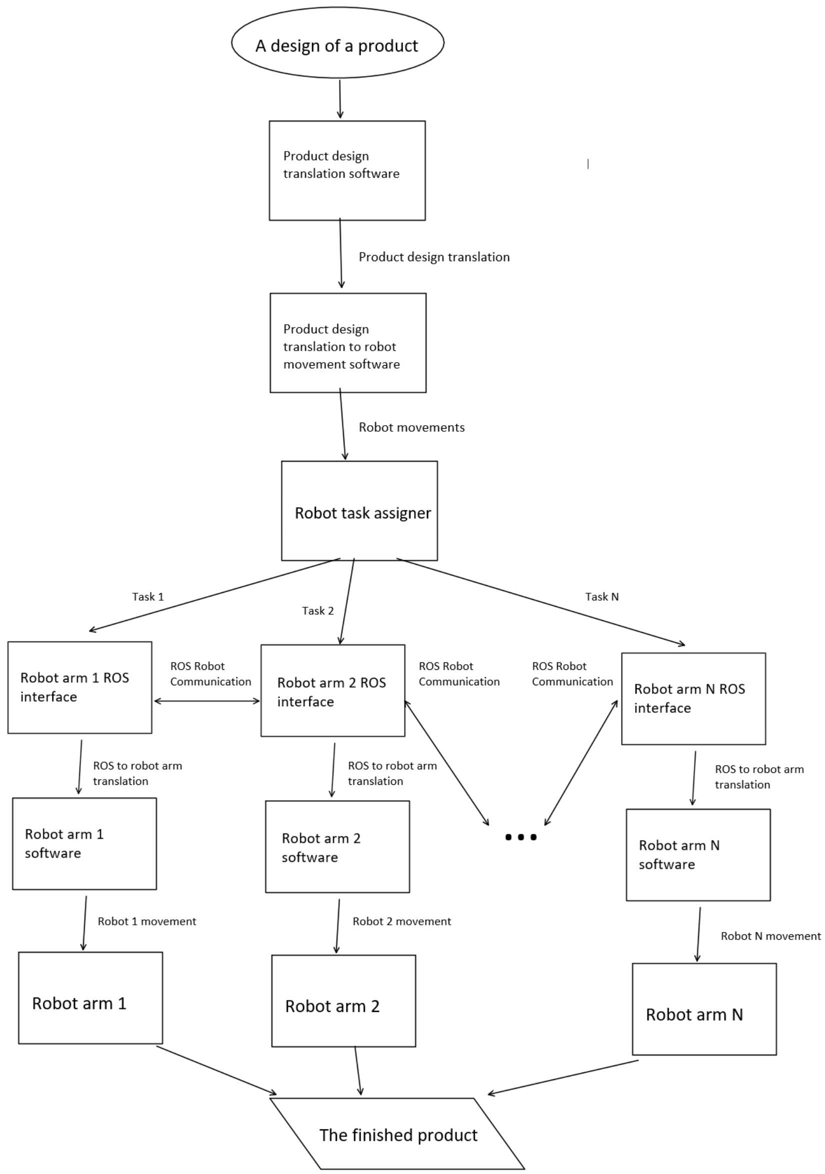


AMBOTS

This design shows the implementation steps of the product design to robot arm movement translation. The robot arms do not understand the design of the product, so a translation layer is needed. The software will extract the necessary information from the product design. It then translates this information into movements for the robotic arms. These movements can be divided up into tasks that specific robotic arms can complete. This task assignment is done with the robot task assigner.

The hardware in this design can be easily expanded to however many robots are necessary to reach the desired finished product. Each robot has their own specific software that controls them. Each different software will receive their assigned task and then move the robotic arms accordingly.

Design Iteration 4:

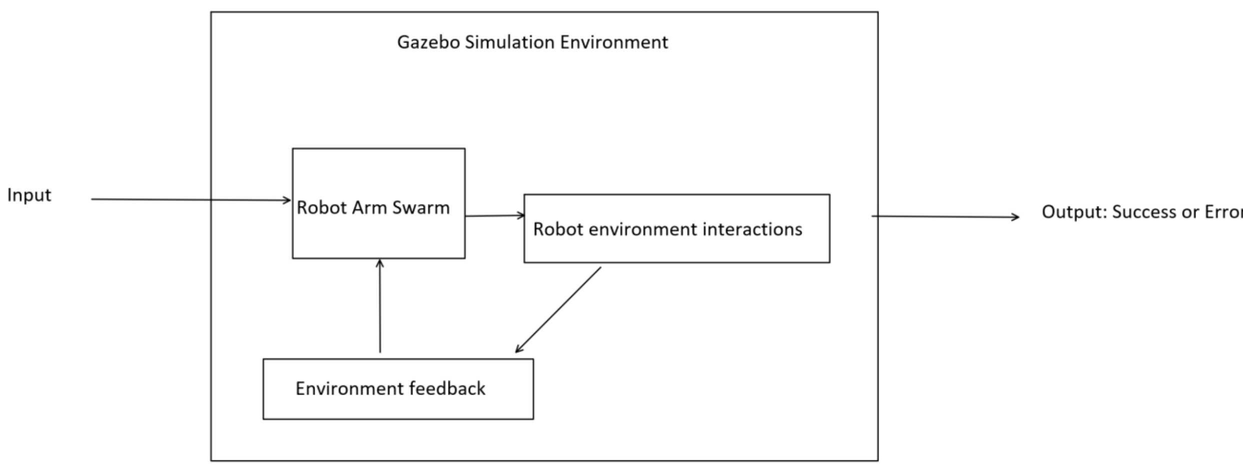


This is our proposed design. On top of the previous design iterations, it integrates ROS 2. This now allows the robot arms to communicate with each other and cooperate on tasks. More complex manufacturing can be done because of the communication between robotic arms such as multicolored 3D prints, complex or intricate machining, along with many other applications. The integration of ROS 2 also allows for different robots to be added into the system easier than before. This is because only a ROS 2 interface with a robot is needed. Any robot with a ROS interface can be added to the swarm of robots and it will be compatible with the system. ROS integration also allows for both the control and simulation of the robots to be done in MoveIt and Gazebo.

Testing Design

The testing will be simulated in Gazebo. This allows us to see how the robotic arms will interact with the environment. The feedback from the environment will get fed back into the swarm of robotic arms. The swarm or operator of the swarm will decide on what do to based on the environment feedback received. The simulation will help us see if the input given to the swarm will be successful or fail without having to test it in the real world.

Testing Diagram:



For the testing diagram, the proposed design from above is represented by the box labeled “Robot Arm Swarm”.

4.3 Risks

Risk	Risk Reduction
Network Security	Using an isolated network that is not connected to the internet along with utilizing encryption and/or a firewall.
Physical Safety	Practice general lab safety and have a kill switch

4.4 Tasks

1. Research ROS, MoveIt, and Gazebo
2. Research Robotic arm options and select 2.
3. Document set up of ROS and MoveIt
4. Research how each robotic arm chosen receives commands
5. Simulate basic movement of robotic arm choices in gazebo
6. Research G-Code
7. Create a program to translate from G-Code file to robotic arms' native language
8. Simulate G-Code movement of robotic arm choices in gazebo
9. Use ROS to communicate between two arms (One moves then tells the other to begin moving)
10. Simulate simultaneous printing with robotic arm choices in gazebo

4.5 Deliverables

- Research: Throughout the development process, research is done on the different software that is to be used. The research will be collected into one coherent document for better understanding of what was learned.
- Documentation: With all the code written, there will be documentation explaining how it works, and why choices were made.
- Software package for ROS2 robotic arm control: This is the collection of software that is used and/or created to complete the goal of simultaneous robotic arm 3D printing.
- Tutorial for setting up and working with the software package: To simplify the process, a complete tutorial will be created for not only the setup of the machine to run the necessary programs, but also the process of running the software produced through this project.
- Final zip file: the final zip file required for the class including a report summarizing the process and implementation of the project along with all of the code written.