16-741 Mechanics of Manipulation - Final Project

Manipulation for Constructing Lincoln Log Structures

Ethan Holand
Robotics Institute
Carnegie Mellon University
Pittsburgh, PA
eholand@andrew.cmu.edu

Morgan Mayborne
Robotics Institute
Carnegie Mellon University
Pittsburgh, PA
mmayborne@andrew.cmu.edu

Fig. 1. Sample Lincoln Logs structures.

***Abstract*—In this project, we explore manipulation within the lens of a classic children's toy: Lincoln Logs. The core element of this system are the logs, which are wooden hexagonal prisms with dual-sided grooves at regular intervals. When assembled together, Lincoln Logs can build incredible, modular structures by interlocking logs in a vertical stack to create a structure. In this project, we seek to utilize a robot to grasp and assemble a structure out of these components.**

I. INTRODUCTION

Invented in 1916, Lincoln Logs have helped children build their fine motor skills for over a century. We wanted to investigate how the fine motor control of a robot compares to a simple child's toy. Within the lens of manipulation, Lincoln Logs represent a tangible medium to test pick-and-place motions. Due to the modular nature of the logs, once simple structures can be assembled, more complicated structures are in reach, such as those seen in figure 1. Compared to other modular building toys like Lego, the looser tolerances of the logs provide a unique challenge, as the position of the structure can drift as a tower is assembled higher.

II. OBJECTIVES

Our primary objective for this project was to use a robot manipulator to assemble a basic structure out of Lincoln Logs. We aimed to develop a strategy that eases the assembly of the structure, and could be extended for more complicated builds. We sought to gain hands-on experience by implementing this on a physical system, gaining knowledge on the challenges of interfacing with a robot arm.

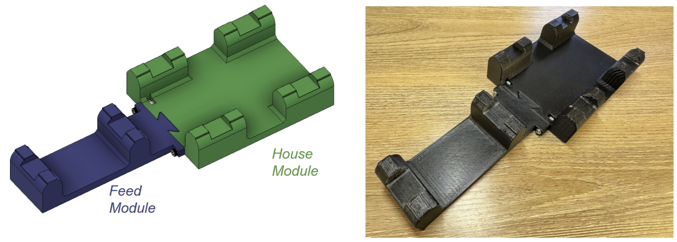

Fig. 2. Custom 3D-printed build platforms to aid pick-and-place operations.

III. METHODOLOGY

A. Build Platform

Initially, we sought to assemble a structure from a cluttered pile of logs, akin to what a child would see when pouring a box onto the ground. However, we quickly realized critical challenges, such as logs obstructed by other logs, or issues with unknown log orientation. We believe that these problems are feasible, but require greater computer vision knowledge, which was outside the scope of the course. Inspired by the pick-and-place platforms demonstrated in the early lectures of the Mechanics of Manipulation course, we altered our strategy to pick logs from a known location. Initially, we considered building larger "silos" of logs from which to pick from. However, due to printing constraints, we simplified this further to a manual feed process.

Our build platform consists of two modules: a feed module and a house module, seen in figure 2. The feed module has two elevated mounting points to hold a two-length log. The house module has a grid of four elevated mounting points for creating a simple square structure. The geometry of these mounting points matched the dimensions of a log. These two modules were printed separately, with a connecting interface at their boundary. This interface consists of a 3D printed dovetail joint and face-mounted M3 screw holes for maximum rigidity. This succeeded in providing a strong connection, requiring us to only calibrate the system once.

B. Pseudo-code

Operating on the constructed build platform, the robot has two central tasks: pick up a log from the feed module and place the log down in the corresponding spot on the house module. Since the robot is interacting with logs that are loaded manually, there was a small consideration for the robot-human interaction. To maintain safety, we ensure that the robot never ran while the human was manually loading the log. An additional safety feature was developing an intermediate point in the pick-and-place sequence for the robot to stay during the manual loading process, to ensure an adequate distance away from the human hand loading the log.

The structure of the program is as follows. Please note for this program that n is the number of layers that we plan to build.

```
FOR i in range(n);
 INCREMENT stack height
 SWAP placement orientation
 FOR j in range(2);
  MOVE robot 20 cm above feeding location
  CHANGE pose
  DELAY 10 seconds while log is fed
  MOVE robot to log location
  GRASP log
  MOVE robot 20 cm above feeding location
  MOVE robot 20 cm above placement location
  CHANGE pose
  MOVE robot to placement location
  RELEASE log
```

C. XML Generator

To achieve the desired motions of our system, we created a custom XML generator in Python to enable programmatic generation of robot trajectories. This provides a simple, human-readable interface for users to chain together "tasks" from a library of motions. Currently, these tasks include a move_to_coor command that takes in a position and orientation and conducts the inverse kinematic motion to that position; a grip command that opens or closes the gripper to a specified position; and a delay command that waits for a certain number of seconds. Within the program, these are inserted in a list form. Using the lxml library, a base document with the appropriate preamble is created, and the user's tasks are injected sequentially into the file.

IV. Experiments

This program was then run on the Kinova v3 Arm (depicted in figure 3), found in the JP Morgan Chase & Co. AI Makerspace in the Tepper Building on Carnegie Mellon's campus. In order to run the program itself, a sequence of coordinates was given - Cartesian x, y, and z as well as relative angles θ_x, θ_y, and θ_z. Initially, the coordinates were found with manual teleoperation, finding points necessary for developing the first two layers of the Lincoln Log structure. With the key coordinates known for pick-and-place actions, as well as the distance between layers in the structure, we created a list of trajectories using the custom XML generator. The robot's user interface, which was opened with a connection to the arm's IP address, imported the XML and ran the motions necessary to translate the arm to the desired positions and orientations.

Fig. 3. A Kinova v3 Arm, utilized for this project

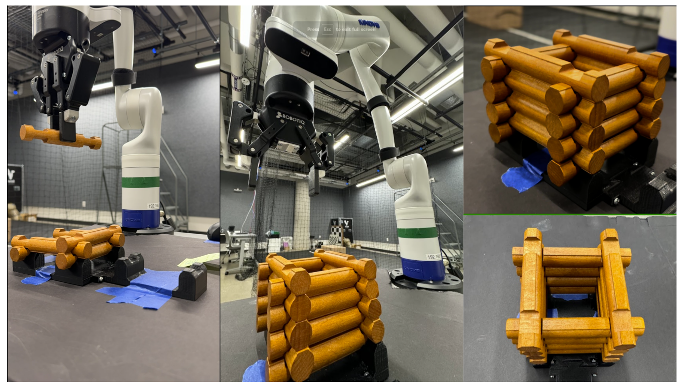

Fig. 4. Left - Image taken during build process. Right - Completed structure.

V. Results

We successfully implemented a method for a robot arm to build a Lincoln Log structure in a 2-by-2 format up to 8 layers, as seen in figure 4. Once the XML generator was developed, the pick-and-place action became quite repeatable with the feeding location and the build location fixed, with no failures in the last 5-10 iterations. The process of finding the coordinates for the feed and build coordinates was also quite intuitive, as the Kinova v3 arm was connected to an Xbox controller that allowed for visual-based feedback and quick teleoperation. Once the first layers were derived in the coordinate space of the robot, it became simple to incorporate the initial measurements into the placements for other layers. It was also quite simple to run programs with the user interface, with the turnaround time for iteration being only a couple of minutes. In summary, the development cycle for the current iteration of the robot was intuitive, and it was simpler than expected to implement.

Comparing our current iteration of the Lincoln Log structure program to our initial vision, there were some difficulties in pick-and-place actions that were not foreseen. Primarily, it was more difficult than expected, particularly time-wise, to develop a grasping protocol for log orientations that are stochastically varied in a pile. Though it may be simple for a child to identify how to grasp a random log, the robot likely would need a trained model to identify a single log in the pile, as well as an additional antipodal grasp solver. Even a "silo" method

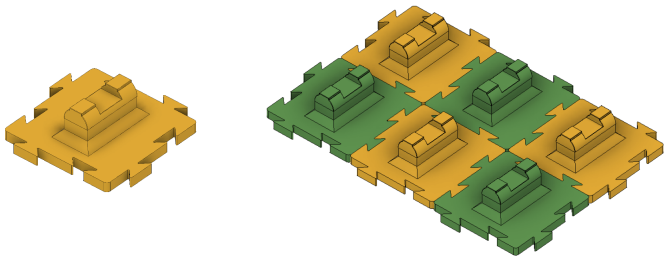

Fig. 5. Modular platform concept. Houses with varying footprints could be constructed by arranging the puzzle pieces.

of maintaining a stack of logs in a known place was more nuanced than first believed, as the robot was shared in a space and session-to-session position repeatability was a difficult challenge.

Another element that was proposed is a procedural structure generator, which would comprise of a python file that would develop a more complex structure than the one devised for this project. This structure generator was proposed to have taken a parameter input and outputted a possible structure. Although a layer-by-layer mapping system was developed when deciding on methodology, limitations with 3D print build volume limited structures to a 2X2 platform, which offer little structural creativity. Later suggestions to create a modular baseplate system would provide a larger base to implement this on.

VI. Future Work

While we were able to achieve our goal of assembling a structure out of Lincoln Logs, there are many more stretch goals we did not implement that could be the subject of future work. A simple near-term goal would be to extend this framework to build larger houses utilizing the three-length logs. This would require a larger feed module with three mounting points instead of two. Additionally, it would require a new house module that has more mounting points. One idea that may be implemented here is a modular puzzle-piece platform design that allows rigid platforms of various shapes and sizes to be assembled and interlocked together, as seen in figure 5.

An additional goal would be to utilize the decorative components in the Lincoln Logs portfolio, such as doors, windows, and roofs. If feed modules are created for these, they could be assembled in a similar manner to the logs.

There are several improvements that could be made to the picking aspect of this project. As previously mentioned, rather than a manual feed process, which requires continual human intervention, a silo could be utilized to stack sets of logs in a known position. As logs are depleted from these reserves, the gripper would move slightly lower to compensate. While these silos could also be loaded manually, a computer vision pipeline could enable a robot to sort a cluttered pile of components into these silos, preparing for a build. A more robust pipeline may even be able to assemble directly from the pile into a finished structure.

With the XML generator we created, programmatic generation of motions was simplified to a set of short commands. In our implementation, the robot ran on a fixed set of motions to build a specific house structure. In the future, a generator could be created that creates a unique house plan given a set of instructions or constraints, such as a desired house shape or an inventory of components.

VII. Conclusion

In this project, we successfully utilized a pick-and-place methodology to assemble a simple structure using standard Lincoln Logs. Completing this task required the design of a custom 3D printed baseplate, programming of a custom trajectory generator, and experimentation with a real-life system. Once set up, the robot repeatedly assembled the structure without failure. There are many future directions to take this work, involving larger structures, different length logs and decorations, and cluttered object picking. While we could not implement all these features in our project, we are glad to have had the opportunity to gain this hands-on manipulation experience with a real robot arm.